# C++ പ്രോഗ്രാമിങ് - ഒരു ആമുഖം

## പ്രധാന ആശയങ്ങൾ

- C++ -ലെ ക്യാരക്ടർ സെറ്റ്
- ടോക്കണുകൾ
  - കീവേഡുകൾ
  - ഐഡന്റിഫയറുകൾ
  - ലിറ്ററലുകൾ
  - പങ്ചുവേറ്ററുകൾ
  - ഓപ്പറേറ്ററുകൾ
- ഇന്റഗ്രേറ്റഡ് ഡവലപ്മെന്റ് എൻവയോൺമെന്റ് (IDE)
  - ജിനി (IDE)

S3E3B6

Bjarne Stroupstrup വികസിപ്പിച്ച ശക്തവും ജനപ്രിയവുമായ ഒരു ഒബ്ജക്ട് ഓറിയന്റഡ് പ്രോഗ്രാമിങ് ഭാഷയാണ് C++ (ഉച്ചരിക്കുന്നത് c പ്ലസ് പ്ലസ്). C++ എന്ന ആശയം വന്നത് C-യോടൊപ്പം + + ഓപ്പറേറ്റർ കൂടി ചേർന്നാണ്. അങ്ങനെ C++ എന്നത് C ഭാഷയുടെ ഒരു വിപുലീകരിച്ച രൂപമായിത്തീർന്നു.

നാം കഴിഞ്ഞ അധ്യായത്തിൽ ചർച്ച ചെയ്ത ക്രമപ്പെടുത്തൽ, തിരഞ്ഞെടുക്കൽ, ആവർത്തനം തുടങ്ങിയ വിവിധ തരം പ്രോഗ്രാമിങ് ആശയങ്ങൾ പ്രാവർത്തികമാക്കാൻ C++ ഭാഷ ഉപയോഗിക്കുന്നു. ഈ അധ്യായത്തിൽ, C++ -ന്റെ അടിസ്ഥാന ആശയങ്ങളെക്കുറിച്ച് ഒരു അവലോകനം നടത്താം. C++ പ്രോഗ്രാം എഴുതുന്നതിനുള്ള വിവിധ ഭാഷാ പ്രൊസസ്സർ പാക്കേജുകളും നമുക്ക് പരിചയപ്പെടാം.

ഏതൊരു ഭാഷയേയും പോലെ, C++ ഭാഷാപഠനവും തുടങ്ങുന്നത് അതിലെ അടിസ്ഥാന ചിഹ്നങ്ങളായ അക്ഷരങ്ങൾ പരിചയപ്പെട്ടുകൊണ്ടാണ്. അതിനുശേഷം വാക്കുകൾ, വാക്യങ്ങൾ (എക്സ്പ്രഷനുകൾ), പ്രസ്താവനകൾ തുടങ്ങിയവയിലൂടെ പഠനക്രമം തുടരും. അക്ഷരങ്ങൾ പഠിച്ച് കൊണ്ട് നമുക്ക് തുടങ്ങാം.

## 5.1 ക്യാരക്ടർ സെറ്റ് (Character set)

നമുക്ക് അറിയാവുന്നത് പോലെ ഇംഗ്ലീഷ്, മലയാളം, ഹിന്ദി തുടങ്ങിയ ഏതൊരു ഭാഷാപഠനവും ആരംഭിക്കുന്നത് അക്ഷരമാലയിലാണ്. അതുപോലെ C++ ഭാഷയ്ക്കും അതിന്റേതായ അക്ഷരമാലയുണ്ട്. ഒരു പ്രോഗ്രാമിങ് ഭാഷയുടെ അക്ഷരമാലയെ അതിന്റെ ക്യാരക്ടർ സെറ്റ് എന്ന് വിളിക്കുന്നു. ഭാഷയിലെ അംഗീകരിക്കപ്പെട്ട ചിഹ്നങ്ങളുടെ ഗണമാണ് അത്. അതിൽ അക്ഷരങ്ങളും, അക്കങ്ങളും മറ്റ് ചിഹ്നങ്ങളും ഉൾപ്പെടുന്നു.

*ന്യൂജഴ്സിയിലെ (USA) മുറേഹില്ലിലുള്ള, AT&T ബെൽ ലബോറട്ടറിയിലാണ് Bjarne Stroustrup C++ വികസിപ്പിച്ചത്. ഇപ്പോൾ അദ്ദേഹം കൊളംബിയ സർവകലാശാലയിലെ സന്ദർശക അധ്യാപകനും ടെക്സാസ് A&M സർവകലാശാലയിലുള്ള കോളേജ് ഓഫ് എഞ്ചിനിയറിംഗിലെ കമ്പ്യൂട്ടർ ശാസ്ത്ര വിഭാഗത്തിന്റെ ചുമതലയുള്ളയാളുമാണ്. അദ്ദേഹത്തിന് നിരവധി പുരസ്കാരങ്ങൾ ലഭിച്ചിട്ടുണ്ട്. C++ ന്റെ ആദ്യകാല നാമം, ക്ലാസ്സുകളോട് കൂടിയ C എന്നായിരുന്നു. പിന്നീട് 1983-ൽ C++ എന്ന് പുനർനാമകരണം ചെയ്തു.*

*Bjarne Stroustrup*

C++ അക്ഷരമാല ചുവടെ ചേർക്കുംവിധം തരം തിരിച്ചിരിക്കുന്നു.

| | | | |
|---|---|---|---|
| (i) | അക്ഷരങ്ങൾ | – | A മുതൽ Z വരെ<br>a മുതൽ z വരെ |
| (ii) | അക്കങ്ങൾ | – | 0 മുതൽ 9 വരെ |
| (iii) | പ്രത്യേക ചിഹ്നങ്ങൾ | – | + - * / ^ \ ( ) [ ] { } = < > . ‘ “ $ , ; : % ? _ # @ |
| (iv) | വൈറ്റ് സ്പെയ്സുകൾ | – | സ്പെയ്സ് ബാർ, ഹൊറിസോണ്ടൽ ടാബ് (→), ക്യാരിയേജ് റിട്ടേൺ ↵, ന്യൂ ലൈൻ, ഫോം ഫീഡ് തുടങ്ങിയവ കൊണ്ട് ഉണ്ടാക്കുന്ന വൈറ്റ് സ്പെയ്സുകൾ |
| (v) | മറ്റു ചിഹ്നങ്ങൾ | – | 256 ASCII ചിഹ്നങ്ങളിൽ ഏതിനേയും C++ ലിറ്ററൽ ആയോ ഡാറ്റ ആയോ പ്രക്രിയയ്ക്കു വിധേയമാക്കാൻ സാധിക്കും. |

*അടുത്തടുത്ത വാക്കുകളും സംഖ്യകളും വേർതിരിക്കാൻ ഉപയോഗിക്കുന്ന അക്ഷരങ്ങളാണ് വൈറ്റ് സ്പെയ്സുകൾ.*

## 5.2 ടോക്കണുകൾ (Tokens)

അക്ഷരമാല പഠിച്ചതിന് ശേഷം അക്ഷരങ്ങൾ ചേർത്തുണ്ടാകുന്ന വാക്കുകളെക്കുറിച്ചുള്ള പഠനമാണ് അടുത്തത്. സ്വാഭാവിക ഭാഷയിലെ വാക്കുകൾക്ക് സമാനമാണ് C++ ഭാഷയിലെ ‘ടോക്കൺ’. ഒരു പ്രോഗ്രാം വികസിപ്പിക്കുന്നതിനുള്ള അടിസ്ഥാന ഘടകങ്ങളാണ് ടോക്കണുകൾ. അവ ലെക്സിക്കൽ യൂണിറ്റുകൾ എന്നും അറിയപ്പെടുന്നു. C++ -ൽ താഴെപ്പറയും വിധം അഞ്ച് തരത്തിലുള്ള ടോക്കണുകളുണ്ട്.

1 കീവേർഡുകൾ
2 ഐഡന്റിഫയറുകൾ
3 ലിറ്ററലുകൾ
4 പഞ്ചുവേറ്ററുകൾ
5 ഓപ്പറേറ്ററുകൾ

### 5.2.1 കീവേഡുകൾ (Keywords)

ഭാഷാ കംബയിലറിന് ഒരു പ്രത്യേക അർഥം നൽകുന്ന വാക്കുകളാണ് (ടോക്കണുകളാണ്) കീവേഡുകൾ. പ്രത്യേക കാര്യങ്ങൾക്കായി ഭാഷ മാറ്റിവച്ച വാക്കുകളായതിനാലും മറ്റാവശ്യങ്ങൾക്ക് പുനർനിർവചിക്കാൻ സാധിക്കാത്തതിനാലും ഇവയെ കീവേഡുകൾ എന്നു വിളിക്കുന്നു. C++ -ലെ 48 കീവേഡുകൾ പട്ടിക 5.1-ൽ അക്ഷരമാലാക്രമത്തിൽ രേഖപ്പെടുത്തിയിട്ടുണ്ട്. അവയുടെ ഉപയോഗം പിന്നീട് വിശദീകരിക്കാം.

| | | | | | |
|---|---|---|---|---|---|
| asm | continue | float | new | signed | try |
| auto | default | for | operator | sizeof | typedef |
| break | delete | friend | private | static | union |
| case | do | goto | protected | struct | unsigned |
| catch | double | if | public | switch | virtual |
| char | else | inline | register | template | void |
| class | enum | int | return | this | volatile |
| const | extern | long | short | throw | while |

*പട്ടിക 5.1 : C++ -ലെ കീവേഡുകൾ*

### 5.2.2 ഐഡന്റിഫയറുകൾ (Identifiers)

സ്ഥലങ്ങൾ, വ്യക്തികൾ, വസ്തുക്കൾ എന്നിവ തിരിച്ചറിയുന്നതിനായി നാം പേരുകൾ നൽകാറുണ്ട്. ഇതിനു വേണ്ടിയാണ് C++ -ൽ ഐഡന്റിഫയറുകൾ ഉപയോഗിക്കുന്നത്. മെമ്മറി സ്ഥാനങ്ങൾ, വാചകങ്ങൾ, ഫങ്ഷനുകൾ, ഒബ്ജക്ടുകൾ, ക്ലാസ്സുകൾ തുടങ്ങിയ പ്രോഗ്രാമിലെ ഘടകങ്ങൾക്ക് പേര് നൽകുന്നതിന് ഉപയോക്താവ് നിർവചിക്കുന്ന വാക്കുകളാണ് ഐഡന്റിഫയറുകൾ. മെമ്മറി സ്ഥാനങ്ങളുടെ ഐഡന്റിഫയറുകളെ വേരിയബിളുകൾ എന്ന് വിളിക്കുന്നു. വാചകങ്ങൾക്ക് നൽകുന്ന ഐഡന്റിഫയറുകളെ ലേബലുകൾ എന്ന് വിളിക്കുന്നു. ഒരു കൂട്ടം പ്രസ്താവനകൾ പ്രതിനിധാനം ചെയ്യുന്ന ഐഡന്റിഫയറുകളാണ് ഫങ്ഷൻ നാമങ്ങൾ.

ഒരു പ്രോഗ്രാമിൽ ഐഡന്റിഫയറുകൾ നിർമ്മിക്കുമ്പോൾ, ചില നിബന്ധനകൾ കൃത്യമായി പാലിക്കേണ്ടതാവശ്യമാണ്. ആ നിയമങ്ങൾ താഴെപ്പറയും വിധമാണ്.

- അക്ഷരങ്ങൾ, അക്കങ്ങൾ, അണ്ടർ സ്കോർ (_) തുടങ്ങിയവയുടെ ഒരു ശ്രേണിയാണ് ഐഡന്റിഫയർ.
- ഐഡന്റിഫയർ തുടങ്ങുന്നത് അക്ഷരത്തിലോ, അണ്ടർ സ്കോറിലോ ( _ ) ആയിരിക്കണം.
- വൈറ്റ് സ്പെയ്സോ പ്രത്യേക ചിഹ്നങ്ങളോ അനുവദനീയമല്ല.
- കീവേഡുകൾ ഐഡന്റിഫയറായി ഉപയോഗിക്കാൻ പാടില്ല.
- ചെറിയ അക്ഷരങ്ങളും വലിയ അക്ഷരങ്ങളും വ്യത്യസ്തമായാണ് C++-ൽ ഉപയോഗിക്കുന്നത്. അതായത് C++ കെയ്സ് സെൻസിറ്റീവാണ്.

സാധുവായ ചില ഐഡന്റിഫയറുകൾക്ക് ഉദാഹരണം: `Count, sumof2numbers, Average_Height, _1stRank, Main, FOR`

താഴെപ്പറയുന്നവ ചില അസാധുവായ ഐഡന്റിഫയറുകളാണ്. അതിനുള്ള കാരണങ്ങളും സൂചിപ്പിച്ചിരിക്കുന്നു.

| | | |
|---|---|---|
| Sum of digits | → | ശൂന്യസ്ഥലങ്ങൾ ഉപയോഗിച്ചിരിക്കുന്നു |
| 1styear | → | അക്കം ആദ്യത്തെ അക്ഷരമായി ഉപയോഗിച്ചിരിക്കുന്നു |
| First . Jan | → | പ്രത്യേക ചിഹ്നം (.) ഉപയോഗിച്ചിരിക്കുന്നു |
| for | → | അതൊരു കീവേഡാണ് |

നമുക്ക് ചെയ്യാം

താഴെ തന്നിരിക്കുന്നവയിൽ നിന്ന് അസാധുവായ ഐഡന്റിഫയറുകൾ കണ്ടെത്തുക. അതിനുള്ള കാരണവും എഴുതുക.

`Data_rec, _data, 1data, data1, my.file, asm, switch, goto, break`

## 5.2.3 ലിറ്ററലുകൾ (Literals)

11-ാം ക്ലാസ് വിദ്യാർഥികൾക്കായുള്ള ഏകജാലക പ്രവേശന സംവിധാനം പരിഗണിക്കുക. പ്രവേശന ഫോറത്തിൽ നിങ്ങൾ ജനനത്തീയതി നൽകിയിട്ടുണ്ടാകും. ജനനത്തീയതി ഒരിക്കലും മാറുന്നില്ല. ഗണിത ശാസ്ത്രത്തിൽ $\pi$ ഒരു സ്ഥിരാംഗവും ഗുരുത്വാകർഷണ സ്ഥിരാംഗം 'g' യുടെ വില (9.8 മീ./സെ.) ഒരിക്കലും മാറാത്തതുമാണെന്ന് നമുക്കറിയാം. അതുപോലെ, C++ -ൽ ഡാറ്റാ ഇനങ്ങളെ പ്രതിനിധാനം ചെയ്യാൻ നാം ഉപയോഗിക്കുന്ന ടോക്കണുകളായ ലിറ്ററലുകൾ, പ്രോഗ്രാം പ്രവർത്തിച്ച് തീരും വരെ വില മാറ്റം വരാതെ തുടരുന്നു. അവയെ പലപ്പോഴും സ്ഥിരാംഗങ്ങൾ എന്ന് വിളിക്കുന്നു. ലിറ്ററലുകൾ താഴെ പറയും വിധം നാലായി തരം തിരിക്കപ്പെട്ടിരിക്കുന്നു.

1 ഇന്റിജർ ലിറ്ററൽ
2 ഫ്ളോട്ടിംഗ് പോയിന്റ് ലിറ്ററൽ
3 ക്യാരക്ടർ ലിറ്ററൽ
4 സ്ട്രിങ് ലിറ്ററൽ

### ഇന്റിജർ ലിറ്ററലുകൾ (പൂർണസംഖ്യാ സ്ഥിരാംഗങ്ങൾ)

1776, 707, -273 എന്നീ സംഖ്യകൾ പരിഗണിക്കുക. അവ പൂർണ ദശാങ്ക വിലകൾ സൂചിപ്പിക്കുന്ന ഇന്റിജർ സ്ഥിരാംഗങ്ങളാണ്. അക്കങ്ങൾ മാത്രം ചേർത്തുണ്ടായിട്ടുള്ള ടോക്കണുകളായ ഇന്റിജർ ലിറ്ററലുകൾ അവിഭാജ്യ ഘടകങ്ങളില്ലാത്ത പൂർണ സംഖ്യകളാണ്. ഇന്റിജർ ലിറ്ററലുകളുടെ സ്വഭാവ സവിശേഷതകൾ താഴെപ്പറയുന്നവയാണ്.

- ഒരു പൂർണ സംഖ്യാ സ്ഥിരാംഗത്തിന് ഒരു അക്കമെങ്കിലും ഉണ്ടായിരിക്കണം. ദശാംശം ഉണ്ടാകരുത്.

നമുക്ക് ചെയ്യാം

*താഴെത്തന്നിട്ടുള്ളവ സാധുവോ അസാധുവോ ആയ പൂർണ്ണ സംഖ്യാ സ്ഥിരാംഗങ്ങളായി തരംതിരിക്കുകയും അവ അസാധുവായതിന്റെ കാരണം കണ്ടെത്തുകയും ചെയ്യുക.*

| | | | | |
|---|---|---|---|---|
| 77,000 | 70 | 314. | -5432 | +15346 |
| +23267 | -.7563 | -02281+0 | 1234E56 | -9999 |

C++ -ൽ ദശാംശ സംഖ്യകൾ (ബേസ്-10) കൂടാതെ അഷ്ടാംഗസംഖ്യകളും (ബേസ്-8) ഷോഡശ സംഖ്യകളും (ബേസ്-16) ലിറ്ററലുകളായി (സ്ഥിരാംഗ ങ്ങളായി) ഉപയോഗിക്കുന്നു. അഷ്ടാംഗ സംഖ്യയെ സൂചിപ്പിക്കുന്നതിന് 0 (പൂജ്യം എന്ന ചിഹ്നം) മുന്നിൽ നൽകുകയും ഷോഡശ സംഖ്യയെ സൂചിപ്പിക്കുന്നതിന് 0x (പൂജ്യം, x) മുന്നിൽ നൽകുകയും ചെയ്യുന്നു. ഉദാഹരണമായി പൂർണസംഖ്യാ സ്ഥിരാംഗ ങ്ങളായ 75 (ബേസ്-10)-ഉം 0113 (ബേസ്-8)-ഉം 0 x 4B (ബേസ്-16)-ഉം എല്ലാം പരസ്പരം തുല്യമാണ്.

- സംഖ്യ പോസിറ്റീവ് എന്നോ നെഗറ്റീവ് എന്നോ കാണിക്കുന്ന + അല്ലെങ്കിൽ – ചിഹ്നം ആദ്യ അക്ഷരമായി വരാവുന്നതാണ്.
- ചിഹ്നം ഇല്ലാത്ത സംഖ്യ പോസിറ്റീവ് സംഖ്യയായി കണക്കാക്കുന്നു.
- മറ്റ് ചിഹ്നങ്ങൾ അനുവദിക്കില്ല.

### ഫ്ളോട്ടിങ് പോയിന്റ് (അസ്ഥിര ദശാംശ സംഖ്യ) ലിറ്ററലുകൾ

3.1459, $3.0\text{x}10^8$, $1.6\text{x}10^{-19}$, 3.0 തുടങ്ങിയ സംഖ്യകൾ നിങ്ങൾക്ക് പരിചിതമാണല്ലോ. ഇവ സാധുവായ നാല് വിലകളാണ്. ഒന്നാമത്തേത് $\pi$ (പൈ)-യുടെ വിലയും, രണ്ടാമത്തേത് പ്രകാശ വേഗത മീറ്റർ/സെക്കന്റിൽ ഉള്ളതും, മൂന്നാമത്തേത് ഇലക്ട്രോണിന്റെ വൈദ്യുത ചാർജും (വളരെ ചെറിയ സംഖ്യ) അവസാനത്തേത് 3 എന്ന സംഖ്യയെ അസ്ഥിര ദശാംശസംഖ്യ ലിറ്ററലായി പ്രസ്താവിച്ചിട്ടുള്ളതുമാണ്.

**അസ്ഥിര ദശാംശസംഖ്യ ലിറ്ററലുകൾ** അവിഭാജ്യ ഘടകങ്ങളുള്ള രേഖീയ സ്ഥിരാംഗങ്ങൾ എന്നും അറിയപ്പെടുന്നു. ഇവ അവിഭാജ്യ രൂപമായോ കൃത്യങ്കരൂപമായോ എഴുതാൻ സാധിക്കും.

രേഖീയ സ്ഥിരാംഗങ്ങൾ അവിഭാജ്യ രൂപത്തിൽ ചിഹ്നമുള്ളതോ ഇല്ലാത്തതോ ആയതും, അക്കങ്ങൾ ചേർന്നതും, അവയ്ക്കിടയിൽ ദശാംശമുള്ളവയുമായിരിക്കും. രേഖീയ സ്ഥിരാംഗങ്ങൾ അവിഭാജ്യ രൂപത്തിലെഴുതുന്നതിനുള്ള നിയമങ്ങൾ ചുവടെ ചേർക്കുന്നു.

- ഒരു രേഖീയ സ്ഥിരാംഗത്തിന് അതിന്റെ അവിഭാജ്യ രൂപത്തിൽ ഒരക്കവും ഒരു ദശാംശവും എങ്കിലും ഉണ്ടായിരിക്കണം.
- അതിന് ഒന്നുകിൽ + (അധികം) അല്ലെങ്കിൽ – (ന്യൂനം) ചിഹ്നം തുടക്കത്തിൽ ഉണ്ടാകാം.
- ചിഹ്നം ഇല്ലാത്ത ഒരു സ്ഥിരാംഗത്തെ അധിക ചിഹ്നമുള്ള സംഖ്യയായി കണക്കാക്കാം.

കൃത്യങ്കരൂപത്തിൽ ഒരു രേഖീയ സ്ഥിരാംഗത്തിന് രണ്ട് ഭാഗങ്ങളുണ്ടാകും – അപൂർണ സംഖ്യാ

ഭാഗവും (മാന്റിസ്സ) കൃത്യങ്കവും. ഉദാഹരണമായി, 5.8 എന്നത് $0.58 \times 10^1 = 0.58$ E1 എന്നെഴുതാനാകും. ഇതിൽ കൃത്യങ്കഭാഗം 1-ഉം (E ക്ക് ശേഷം വരുന്ന ഭാഗം) മാന്റിസ്സ ഭാഗം 0.58-ഉം (E യ്ക്ക് മുമ്പുള്ള ഭാഗം) ആണ്. E 1 എന്നത് $10^1$-നെ സൂചിപ്പിക്കുന്നു. കൃത്യങ്ക രൂപത്തിൽ രേഖീയ സ്ഥിരാംഗം എഴുതുന്നതിനുള്ള നിയമങ്ങൾ താഴെപ്പറയും വിധമാണ്.

- കൃത്യങ്ക രൂപത്തിൽ രേഖീയ സ്ഥിരാംഗങ്ങൾക്ക് രണ്ട് ഭാഗങ്ങളുണ്ടായിരിക്കും. ഒരു പൂർണ സംഖ്യാ ഭാഗവും ഒരു കൃത്യങ്കവും.
- അപൂർണ സംഖ്യാ ഭാഗം ഒന്നുകിൽ ഒരു പൂർണ സംഖ്യാ രൂപത്തിലോ അല്ലെങ്കിൽ സാധുവായ അവിഭാജ്യ രൂപത്തിലോ ആയിരിക്കും.
- അപൂർണ സംഖ്യാ ഭാഗത്തെ തുടർന്ന് E അല്ലെങ്കിൽ e എന്ന അക്ഷരവും കൃത്യങ്കവുമുണ്ടാകും.
- കൃത്യങ്കം പൂർണസംഖ്യ ആയിരിക്കണം.

താഴെത്തന്നിട്ടുള്ളവ സാധുവായ രേഖീയ സ്ഥിരാംഗങ്ങളാണ്.

| | | | |
|---|---|---|---|
| 52.0 | 107.5 | -713.8 | -.00925 |
| 453.E-5 | 1.25E08 | .212E04 | 562.0E09 |
| 152E+8 | 1520E04 | -0.573E-7 | -.097 |

ചില അസാധുവായ രേഖീയ സ്ഥിരാംഗങ്ങൾ കാരണസഹിതം ഇതോടൊപ്പം ചേർക്കുന്നു.

58,250.262 (കോമ ഉപയോഗിച്ചിരിക്കുന്നു), 5.8E (കൃത്യങ്ക ഭാഗം ഇല്ല), 0.58E2.3 (കൃത്യങ്കമായി അവിഭാജ്യ സംഖ്യ ഉപയോഗിച്ചിരിക്കുന്നു).

നമുക്ക് ചെയ്യാം

*താഴെ തന്നിട്ടുള്ളവയെ കാരണസഹിതം സാധുവോ അസാധുവോ ആയ രേഖീയ സ്ഥിരാംഗങ്ങളായി തരംതിരിക്കുക.*

| | | | | |
|---|---|---|---|---|
| 77, 00,000 | 7.0 | 3.14 | -5.0E5.4 | +53.45E-6 |
| +532.67. | .756E-3 | -0.528E10 | 1234.56789 | 34,56.24 |
| 4353 | +34/2 | 5.6E | 4356 | 0 |

## ക്യാരക്ടർ ലിറ്ററൽ

സാധാരണ ലിംഗഭേദത്തെ കാണിക്കുന്നതിനായി ആൺ (Male) എന്നതിന് 'M' അല്ലെങ്കിൽ 'm' എന്നും പെൺ (Female) എന്നതിന് 'F' അല്ലെങ്കിൽ 'f' എന്നും ഉപയോഗിക്കാറുണ്ട്. അതുപോലെ, അതെ (Yes) എന്നതിന് 'y' അല്ലങ്കിൽ 'Y' എന്നും അല്ല (No) എന്നതിന് 'n' അല്ലെങ്കിൽ 'N' എന്നും നാം ഉപയോഗിക്കുന്നു. ഇവയെല്ലാം ഒറ്റ അക്ഷരങ്ങളാണ്. പ്രോഗ്രാം പ്രവർത്തിച്ച് തീരും വരെ അവ വില മാറ്റമില്ലാതെ തുടരുന്നു. ഒരു ജോഡി ഏക സൂചകങ്ങൾക്കുള്ളിൽ (ഒറ്റ ഉദ്ധരണികൾക്ക് ഉള്ളിൽ) ഉപയോഗിച്ചിരിക്കുന്ന അക്ഷരത്തെ ക്യാരക്ടർ ലിറ്ററൽ അല്ലെങ്കിൽ ക്യാരക്ടർ സ്ഥിരാംഗം എന്നു വിളിക്കുന്നു.

ഉദ്ധരണി ഇല്ലാതെ ഉപയോഗിക്കുന്ന X ഐഡന്റിഫയറും ഉദ്ധരണിക്കുള്ളിൽ ഉപയോഗിക്കുന്ന 'X' ക്യാരക്ടർ ലിറ്ററലുമാണ്. ക്യാരക്ടർ സ്ഥിരാംഗത്തിന്റെ വില എന്നത് അതിന്റെ ASCII വിലയാണ്. 'c' -യുടെ വില എന്നത് 'c' -യുടെ ASCII വിലയായ 99-ഉം 'A'-യുടെ വില എന്നത് അതിന്റെ ASCII വിലയായ 65-ഉം ആണ്.

C++ ഭാഷയിൽ ചില ചിത്രീകരിക്കാനാകാത്ത ക്യാരക്ടർ സ്ഥിരാംഗങ്ങളുണ്ട്, അവയെ കീ ബോർഡിൽ നിന്ന് നേരിട്ട് ടൈപ്പ് ചെയ്യാൻ സാധിക്കില്ല. ഉദാഹരണമായി, ക്യാരിയേജ് റിട്ടേൺ അല്ലെങ്കിൽ എന്റർ കീ, ടാബ് കീ, ബാക്ക് സ്പെയ്സ് കീ എന്നിവ ഒരു രീതിയിലും പ്രകടിപ്പിക്കാൻ സാധ്യമല്ല. ഇത്തരം പ്രകടിപ്പിക്കാനാകാത്ത ചിഹ്നങ്ങളെ പ്രതിനിധീകരിക്കാനായി **എസ്കേപ്പ് സീക്വൻസുകൾ** ഉപയോഗിക്കാം. അവയിൽ ബാക്ക് സ്ലാഷും തുടർന്ന് വരുന്ന ഒന്നോ അതിലധികമോ അക്ഷരങ്ങളും അടങ്ങിയിരിക്കും. എസ്കെയ്പ് സീക്വൻസുകൾ ഒരു ജോഡി ഏക സൂചകങ്ങൾക്കുള്ളിലുള്ള ഒന്നിലധികം അക്ഷരങ്ങൾ ചേർന്നതാണെങ്കിലും ആനുപാതികമായ ഒരു ASCII കോഡുപയോഗിച്ചാണ് അത് സൂചിപ്പിക്കുന്നത് എന്ന കാര്യം ശ്രദ്ധിക്കേണ്ടതാണ്. അതുകൊണ്ടാണ് അവയെ ക്യാരക്ടർ സ്ഥിരാംഗങ്ങളായി കണക്കാക്കുന്നത്. പട്ടിക 5.2-ൽ എസ്കേപ്പ് സീക്വൻസുകളും ആനുപാതികമായ ക്യാരക്ടറുകളും കുറിച്ചിട്ടുണ്ട്.

പട്ടിക 5.2-ൽ \', \", \? എന്നിവ നിങ്ങൾക്ക് കാണാൻ സാധിക്കും. ഈ ചിഹ്നങ്ങൾ (അക്ഷരങ്ങൾ) കീ ബോർഡിൽ നിന്ന് ടൈപ്പ് ചെയ്യാൻ സാധിക്കുമെങ്കിലും എസ്കേപ്പ് സീക്വൻസ് ഇല്ലാതെ ഉപയോഗിച്ചാൽ അവ പ്രത്യേക ഉദ്ദേശ്യവും കാര്യവുമായിരിക്കും നിർവഹിക്കുന്നത്. അവ അതുപോലെ തന്നെ പ്രദർശിപ്പിക്കുകയോ അച്ചടിക്കുകയോ ചെയ്യേണ്ടി വന്നാൽ എസ്കേപ്പ് സീക്വൻസ് ഉപയോഗിക്കണം. ചില സാധുവായ ക്യാരക്ടർ സ്ഥിരാംഗങ്ങൾക്ക് ഉദാഹരണങ്ങളാണ്:

's', 'S', '$', '\n', '+', '9'

ചില അസാധുവായ ക്യാരക്ടർ സ്ഥിരാങ്കങ്ങൾ കാരണ സഹിതം തന്നിരിക്കുന്നു.

A (ഏക ഉദ്ധരണി ഇല്ല), '82' (ഒന്നിലധികം അക്ഷരങ്ങൾ), "K" (ഏക ഉദ്ധരണിക്ക് പകരം ജോഡിയായ ഉദ്ധരണികൾ), '\g' (അസാധുവായ എസ്കേപ്പ് സീക്വൻസ് അല്ലെങ്കിൽ ഒന്നിലധികം അക്ഷരങ്ങൾ).

| എസ്കേപ്പ് സീക്വൻസ് | ആനുപാതികമായ ചിത്രീകരിക്കാനാകാത്ത ചിഹ്നങ്ങൾ |
|---|---|
| `\a` | ഓഡിയബിൾ ബെൽ (അലെർട്ട്) |
| `\b` | ബാക്ക് സ്പെയ്സ് |
| `\f` | ഫോം ഫീഡ് |
| `\n` | ന്യൂലൈൻ അല്ലെങ്കിൽ ലൈൻഫീഡ് |
| `\r` | ക്യാരിയേജ് റിട്ടേൺ |
| `\t` | ഹൊറിസോണ്ടൽ ടാബ് |
| `\v` | വെർട്ടിക്കൽ ടാബ് |
| `\\` | ബാക്ക് സ്ലാഷ് |
| `\'` | സിംഗിൾ ക്വോട്ട് (ഏകസൂചകം) |
| `\"` | ഡബിൾ ക്വോട്ട് (ജോഡിയായ സൂചകം) |
| `\?` | ക്വസ്റ്റ്യൻ മാർക്ക് (ചോദ്യ ചിഹ്നം) |
| `\0` | നൾ ക്യാരക്ടർ |

*പട്ടിക 5.2: എസ്കേപ്പ് സീക്വൻസുകൾ*

ഒക്ടൽ സംഖ്യകളെയും ഹെക്സാ ഡെസിമൽ സംഖ്യകളെയും എസ്കേപ്പ് സീക്വൻസുകളുടെ സഹായത്താലാണ് C++-ൽ സൂചിപ്പിക്കുന്നത്. \0n-ഉം \xHn-ഉം യഥാക്രമം ഒക്ടൽ സംഖ്യാ സമ്പ്രദായത്തിലും ഹെക്സാ ഡെസിമൽ സംഖ്യാന സമ്പ്രദായത്തിലും ഉള്ള സംഖ്യകളെ സൂചിപ്പിക്കുന്നു.

### സ്ട്രിങ് ലിറ്ററൽ

നന്ദന ഒരു വിദ്യാർഥിനിയാണ്. അവൾ ബാപ്പുജി നഗറിൽ താമസിക്കുന്നു. ഇവിടെ "Nandana" എന്നത് ഒരു പെൺകുട്ടിയുടെ പേരും "Bapuji Nagar" എന്നത് ഒരു സ്ഥല നാമവുമാണ്. ഈ രീതിയിലുള്ള ഡാറ്റ, പ്രോഗ്രാമുകളുടെ സഹായത്താൽ പ്രോസസ്സ് ചെയ്യേണ്ടി വരും. അത്തരം ഡാറ്റയെ സ്ട്രിങ് സ്ഥിരാംഗങ്ങളായി പരിഗണിക്കുകയും ജോഡികളായ ഉദ്ധരണികൾക്കുള്ളിൽ ഉൾപ്പെടുത്തുകയും ചെയ്യുന്നു. തുടർച്ചയായ ഒന്നോ അതിലധികമോ അക്ഷരങ്ങൾ (ചിഹ്നങ്ങൾ), രണ്ട് ജോടിയായ സൂചകങ്ങൾക്കുള്ളിൽ (ഉദ്ധരണികൾക്കുള്ളിൽ) അടക്കം ചെയ്തതാണ് സ്ട്രിങ് സ്ഥിരാംഗങ്ങൾ. ഉദാഹരണമായി "Hello friends", "123", "C++", "Baby\'s Day Out" എന്നിവ സാധുവായ സ്ട്രിങ് സ്ഥിരാംഗങ്ങളാണ്.

നമുക്ക് ചെയ്യാം

*താഴെ തന്നിട്ടുള്ളവ പല വിഭാഗങ്ങളിലുള്ള ലിറ്ററലുകളായി (സ്ഥിരാംഗങ്ങളായി) തരംതിരിക്കുക.*

```
'a'          'rita'     -124       12.5      -12e-1
"raju\'s pen"   0                 -11.999   '\\'       32760
```

## 5.2.4 പഞ്ചുവേറ്ററുകൾ (പൂർണ വിരാമ ചിഹ്നങ്ങൾ) (Punctuators)

വാക്യങ്ങളിലെ വ്യാകരണത്തിന്റെ പൂർണതയ്ക്കായി പഞ്ചുവേഷൻ അടയാളങ്ങൾ ഇംഗ്ലീഷ്, മലയാളം തുടങ്ങിയ ഭാഷകളിലുപയോഗിക്കുന്നുണ്ട്. ഉദാഹരണമായി, 'ആരാണ് C++ വികസിപ്പിച്ചത്?' എന്ന വാക്യം ശ്രദ്ധിക്കുക. ഇവിടെ '?' എന്നത് പഞ്ചുവേഷൻ അടയാളവും അത് സൂചിപ്പിക്കുന്നത് ഒരു ചോദ്യ പ്രസ്താവനയെയുമാണ്. അതുപോലെ എല്ലാ വാക്യങ്ങളുടെയും അവസാനം നാം പൂർണവിരാമ ചിഹ്നം (.) ഇടുന്നു. അത്തരത്തിൽ അർഥ സംബന്ധമായോ, പദവിന്യാസപരമായോ ഉള്ള പൊരുൾ കംബയിലറിന് എത്തിക്കുവാനായി C++ -ലും ചില പ്രത്യേക ചിഹ്നങ്ങൾ ഉണ്ട്. അവയെ പഞ്ചുവേറ്ററുകൾ എന്ന് വിളിക്കുന്നു.

അവയ്ക്കുള്ള ഉദാഹരണങ്ങളാണ് # ; ' " ( ) { } [ ] എന്നിവ. ഓരോ പഞ്ചുവേറ്ററിന്റെയും ഉപയോഗം പിന്നീട് ചർച്ച ചെയ്യാം.

## 5.2.5 ഓപ്പറേറ്ററുകൾ (Operators)

5-ഉം 3-ഉം തമ്മിൽ കൂട്ടേണ്ടി വരുമ്പോൾ, അവ 5+3 എന്ന് നാം സൂചിപ്പിക്കുന്നു. ഇവിടെ + എന്നത് സങ്കലനം എന്ന പ്രവൃത്തിയെ കാണിക്കുന്ന ഓപ്പറേറ്റർ ആണ്. C++-ൽ ഇതുപോലെയുള്ള ധാരാളം ഓപ്പറേറ്ററുകൾ ഉണ്ട്. കംബയിലറിനോട് ഒരു പ്രത്യേക പ്രവൃത്തിയെക്കുറിച്ച് പറയുന്നതിന് ഉപയോഗിക്കുന്ന ചിഹ്നമാണ് ഓപ്പറേറ്റർ. ഏതെങ്കിലും തരത്തിലുള്ള പ്രവൃത്തിയെ ഉത്തേജിപ്പിക്കുന്ന ടോക്കണുകളാണ് അവ. ഓപ്പറന്റുകൾ എന്ന് വിളിക്കുന്ന ഒരു കൂട്ടം ഡാറ്റാകളിൽ ഓപ്പറേറ്ററുകൾ പ്രയോഗിക്കുന്നു. അരിത്മെറ്റിക്, ലോജിക്കൽ, റിലേഷണൽ, കണ്ടീഷണൽ, അസൈൻമെന്റ് തുടങ്ങിയ വിവിധ തരം ഓപ്പറേറ്ററുകൾ C++ -ൽ ഉണ്ട്. ഓപ്പറേറ്ററുകളെക്കുറിച്ച് കൂടുതലായി നമുക്ക് അടുത്ത അധ്യായത്തിൽ ചർച്ച ചെയ്യാം.

**നമുക്ക് ചെയ്യാം**

*താഴെ തന്നിട്ടുള്ളവ വിവിധയിനം ടോക്കണുകളായി തരംതിരിക്കുക.*

```
/      -24               +      -12e-1    '"KL01"
Sum  "raju\'s pen"   if    rita        '\\'
break    }
```

## 5.3 ഇന്റഗ്രേറ്റഡ് ഡവലപ്മെന്റ് എൻവയോൺമെന്റ് – IDE

ഒരു C++ പ്രോഗ്രാമിലെ അടിസ്ഥാന ഘടകങ്ങൾ നാം ഇപ്പോൾ പഠിച്ചു കഴിഞ്ഞു. C++ പ്രോഗ്രാമുകൾ എഴുതിത്തുടങ്ങുന്നതിന് മുൻപ് എവിടെയാണ് പ്രോഗ്രാം ടൈപ്പ് ചെയ്യേണ്ടതെന്ന് നാം അറിഞ്ഞിരിക്കണം. മറ്റ് പ്രോഗ്രാമിങ് ഭാഷകളെപ്പോലെ ഒരു ടെക്സ്റ്റ് എഡിറ്റർ ഉപയോഗിച്ച് C++ പ്രോഗ്രാം നിർമ്മിക്കാം. C++ പ്രോഗ്രാമുകൾ സൃഷ്ടിക്കാനായി ടർബോ C++, ബോറോലാന്റ് C++, GCC തുടങ്ങിയ കമ്പയിലറുകൾ IDE ലഭ്യമാക്കുന്നു. ഇവയിൽ പല IDE-കളിലും ടൈപ്പിങ്, എഡിറ്റിങ്, സേർച്ചിങ്, ലിങ്കിങ്, കമ്പയിലിങ്, എക്സിക്യൂട്ടിങ് എന്നീ സൗകര്യങ്ങൾ ഉണ്ട്. ഇത്തരം പ്രവർത്തനങ്ങൾക്കായി നാം ഇവിടെ ജിനി IDE (ഐ.ടി.@ സ്കൂൾ ഉബുണ്ടു ലിനക്സ് 14.04) ഉപയോഗിക്കുന്നു.

### GCC യോടൊപ്പമുള്ള ജിനി IDE

ലിനക്സ് ഓപ്പറേറ്റിംഗ് സിസ്റ്റത്തോടൊപ്പം ലഭ്യമാകുന്ന ഒരു സ്വതന്ത്ര സോഫ്റ്റ്‌വെയറാണ് GCC കമ്പയിലർ. GCC എന്നത് GNU കമ്പയിലർ കളക്ഷനെ പ്രതിനിധാനം ചെയ്യുന്നതും ISO C++ മാനദണ്ഡങ്ങൾ അനുസരിച്ച് പ്രവർത്തിക്കുന്നതുമായ ഒരു ജനപ്രിയ കമ്പയിലറാണ്. C++ പ്രോഗ്രാമുകൾ എഴുതുന്നതിനും, കമ്പയിൽ ചെയ്യുന്നതിനും എക്സിക്യൂട്ട് ചെയ്യുന്നതിനുമുള്ള ഒരു ക്രോസ് - പ്ലാറ്റ്ഫോം IDE ആണ് ജിനി.

### A. എഡിറ്റ് വിൻഡോ തുറക്കുന്ന വിധം

ഉബുണ്ടു ലിനക്സിന്റെ ആപ്ലിക്കേഷൻസ് മെനുവിൽ നിന്നുമാണ് ജിനി IDE-യുടെ എഡിറ്റ് വിൻഡോ തുറക്കുന്നത്.

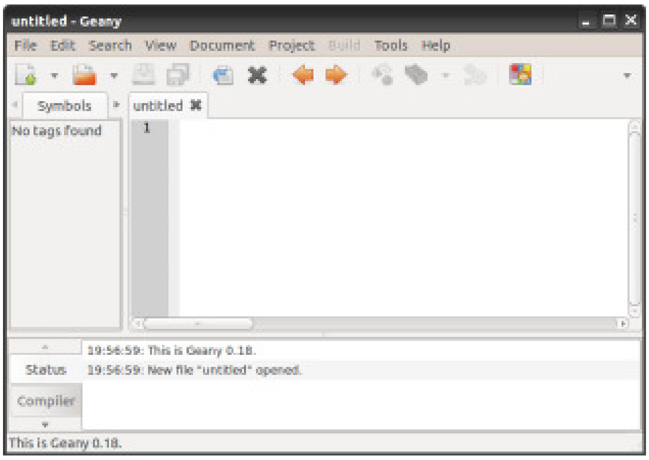
**അപ്ലിക്കേഷനുകൾ → പ്രോഗ്രാമിങ് →ജിനി**

ജിനി IDE അതിന്റെ വിൻഡോ തുറക്കുന്നത് ചിത്രം 5.1-ൽ കാണിച്ചിരിക്കും വിധമാണ്. അതിന് ഒരു ടൈറ്റിൽ ബാർ, മെനു ബാർ, ടൂൾ ബാർ, കോഡ് എഡിറ്റ് ചെയ്യാനുള്ള സ്ഥലം എന്നിവ ഉണ്ടാകും. അൺ ടൈറ്റിൽഡ് എന്ന പേരിലുള്ള ഒരു ടാബും അവിടെ കാണാൻ സാധിക്കും. വിൻഡോസ് ഓപ്പറേറ്റിംഗ് സിസ്റ്റത്തിൽ ജിനി 1.24 ആണ് നാം ഉപയോഗിക്കുന്നതെങ്കിൽ, തുറക്കുന്ന ജാലകം (വിൻഡോ) ചിത്രം 5.2-ൽ കാണിച്ചിരിക്കും വിധമായിരിക്കും. മുകളിൽ പരാമർശിച്ച രണ്ട് ജാലകങ്ങളും (വിൻഡോകളും) സമാനമാണെന്ന് നമുക്ക് കാണാം.

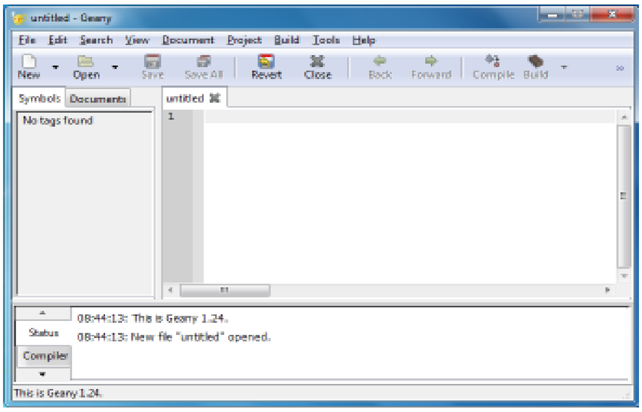
അൺടൈറ്റിൽഡ് എന്ന തനത് പേരുള്ള ഫയലിൽ നമുക്ക് പ്രോഗ്രാം എഡിറ്റ് ചെയ്യാം. ഒരു പുതിയ ഫയൽ തുറക്കാനായി ഫയൽ മെനുവിലെ ന്യൂ ഓപ്ഷൻ തിരഞ്ഞെടുക്കുകയോ ടൂൾ ബാറിലുള്ള ന്യൂ ബട്ടൺ ക്ലിക്ക് ചെയ്യുകയോ ആകാം. ഇതേ ആവശ്യത്തിനായി Ctrl+N എന്നീ കീ-കൾ ഒരുമിച്ചും ഉപയോഗിക്കാം.

## B. പ്രോഗ്രാം സേവ് ചെയ്യുന്നു

ഒരിക്കൽ ഫയൽ തുറന്നാൽ C++ പ്രോഗ്രാം നൽകുകയും അനുയോജ്യമായ ഫയൽ നാമത്തോടൊപ്പം .cpp എന്ന എക്സ്റ്റൻഷൻ കൊടുത്ത് അത് സേവ് ചെയ്യുകയും ആകാം. GCC എന്നത് ഒരു കൂട്ടം കമ്പയിലറുകൾ ആയതിനാൽ കോഡിന്റെ കമ്പയിലേഷന് വേണ്ടി ഏത് കമ്പയിലർ തിരഞ്ഞെടുക്കണമെന്ന് തീരുമാനിക്കുന്നത് അതിന്റെ എക്സ്റ്റൻഷൻ നോക്കിയാണ്. അതുകൊണ്ട് ഫയൽ എക്സ്റ്റൻഷൻ കൃത്യമായി നൽകിയിരിക്കണം. പ്രോഗ്രാം ടൈപ്പ് ചെയ്യുന്നതിന് മുൻപായി നാം ഫയൽ നാമം നൽകുകയാണെങ്കിൽ, പ്രോഗ്രാമിലുള്ള വിവിധയിനം ടോക്കണുകളെ വേർതിരിച്ച് കാണിക്കാനായി GCC സ്വയം പല നിറങ്ങൾ ലഭ്യമാക്കും. സോഴ്സ് കോഡിലെ വാചകങ്ങളുടെ സ്ഥാനം തിരിച്ചറിയുന്നതിനുള്ള ഇൻഡന്റേഷനും ഇത് ഉപയോഗിക്കുന്നു. ഇൻഡന്റേഷൻ എന്ന ആശയം നമുക്ക് പിന്നീട് ചർച്ച ചെയ്യാം.




*ചിത്രം 5.1: ഉബുണ്ടു ലിനക്സിലെ ജിനി IDE- - യുടെ പ്രാരംഭ സ്ക്രീൻ*



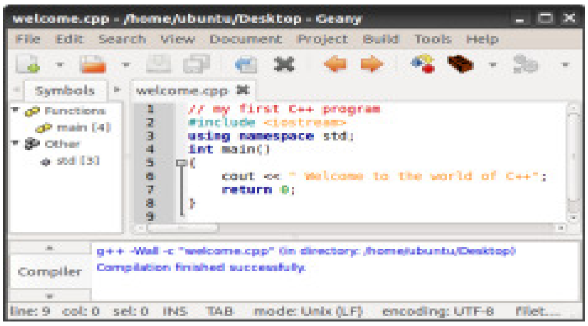

*ചിത്രം 5.2: വിൻഡോസ് ഓപ്പറേറ്റിങ് സിസ്റ്റത്തിൽ*

പ്രോഗ്രാം 5.1-ൽ കാണിച്ചിരിക്കുന്നത് പോലെ ലളിതമായ ഒരു പ്രോഗ്രാം നമുക്ക് എഴുതുകയും welcome.cpp എന്ന പേരിൽ അത് സേവ് ചെയ്യുകയും ആകാം.

**പ്രോഗ്രാം 5.1: IDE മനസ്സിലാക്കുന്നതിനുള്ള ഒരു പ്രോഗ്രാം.**

```
// my first C++ program
#include<iostream>
using namespace std;
int main()
{
    cout << "Welcome to the world of C++";
    return 0;
}  //end of program
```

പ്രോഗ്രാം 5.1 കൊടുത്തശേഷമുള്ള IDE വിൻഡോ ചിത്രം 5.3-ൽ കാണിച്ചിരിക്കുന്നു. ടോക്കണുകളിൽ ഉപയോഗിച്ചിട്ടുള്ള വിവധ നിറങ്ങൾ നിരീക്ഷിക്കുക.



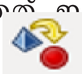

*ചിത്രം 5.3: ജിനി IDE-യിൽ ഒരു പേര് നൽകി പ്രോഗ്രാം സേവ് ചെയ്യുന്നു.*

പ്രോഗ്രാം സേവ് ചെയ്യുന്നതിനായി, ഫയൽ മെനുവിലെ സേവ് ഓപ്ഷൻ എടുക്കുകയോ കീബോർഡിലെ Ctrl+S എന്ന കുറുക്കുവഴി ഉപയോഗിക്കുകയോ ചെയ്യുക. ടൂൾ ബാറിലെ സേവ് ബട്ടൺ ക്ലിക്ക് ചെയ്ത് കൊണ്ടും ഫയൽ സേവ് ചെയ്യാവുന്നതാണ്. ഇടയ്ക്കിടെ Ctrl+S അമർത്തിക്കൊണ്ട് പ്രോഗ്രാം സേവ് ചെയ്യുന്നതാണ് ഉചിതമായ രീതി .

അപ്രതീക്ഷിതമായ സിസ്റ്റത്തിന്റെ പിഴവോ, വൈദ്യുതിത്തകരാറോ മൂലം ഡാറ്റ നഷ്ടപ്പെടുന്നത് ഒഴിവാക്കാൻ ഇത് സഹായിക്കുന്നു. ഒരിക്കൽ, പ്രോഗ്രാം ടൈപ്പ് ചെയ്യുന്നത് പൂർണമായാൽ കമ്പയിൻ ചെയ്യുകയോ വ്യത്യാസം വരുത്തുകയോ ചെയ്യുന്നതിന് മുൻപായി ഫയൽ സേവ് ചെയ്യുന്നതാണ് ഏറ്റവും ഉചിതം. താൽക്കാലികമായ അസ്ഥിര പ്രാഥമിക മെമ്മറിയിൽ (വോളട്ടൈൽ പ്രൈമറി മെമ്മറിയിൽ) നിന്ന് ഒരു ഫയൽ ദീർഘകാലത്തേക്കുള്ള സുസ്ഥിരമായ ദ്വിതീയ മെമ്മറിയിലേക്ക് (നോൺ വോളട്ടൈൽ സെക്കന്ററി മെമ്മറിയിലേക്ക്) പകർത്തുന്നതിനെയാണ് പ്രോഗ്രാം സേവ് ചെയ്യുക എന്ന് പറയുന്നത്.

*വിവിധ C++ കമ്പയിലറുകൾക്കനുസരിച്ച് പല എക്സ്റ്റൻഷനുകളാണ് പ്രോഗ്രാം ഫയലുകൾക്ക് നൽകുന്നത്. അതായത് വിവിധ കമ്പയിലറുകൾ വ്യത്യസ്തങ്ങളായ ഫയൽ എക്സ്റ്റൻഷനുകൾ പിൻതുടരുന്നു. ഉദാഹരണമായി .cpp, .cxx, .cc, .c++ തുടങ്ങിയ എക്സ്റ്റൻഷനുകൾ ശ്രദ്ധിക്കുക.*

## C. പ്രോഗ്രാമിന്റെ കമ്പയിലിങ്ങും ലിങ്കിങ്ങും

പ്രോഗ്രാം കമ്പയിൽ ചെയ്യുകയും, തെറ്റ് കണ്ടെത്തിയാൽ തിരുത്തുകയും ചെയ്യുന്ന ഘട്ടമാണ് അടുത്തത്. ഇതിന് ബിൽഡ് മെനുവിലെ കമ്പയിൽ ഓപ്ഷൻ എടുക്കുകയോ കമ്പയിൽ ബട്ടൺ  ഉപയോഗിക്കുകയോ ചെയ്യാം. തെറ്റുകളുണ്ടായാൽ, ആ തെറ്റുകൾ താഴെ ഭാഗത്തുള്ള കമ്പയിലറിന്റെ സ്ഥിതി കാണിക്കുന്ന വിൻഡോയിൽ പ്രദർശിപ്പിക്കുകയും മറിച്ചാണെങ്കിൽ, കമ്പയിലേഷൻ വിജയകരമായി പൂർത്തീകരിച്ചു എന്ന സന്ദേശം പ്രദർശിപ്പിക്കുകയും ചെയ്യുന്നു (ചിത്രം 5.3 പരിശോധിക്കുക). വിജയകരമായ കമ്പയിലേഷന് ശേഷം ലിങ്ക് ചെയ്യാനായി ടൂൾ ബാറിലെ ബിൽഡ് ബട്ടൺ ക്ലിക്ക് ചെയ്യുകയോ ബിൽഡ് മെനുവിലെ ബിൽഡ് ഓപ്ഷൻ തിരഞ്ഞെടുക്കുകയോ ചെയ്യുക. ഇപ്പോൾ പ്രോഗ്രാം എക്സിക്യൂഷന് തയാറായിക്കഴിഞ്ഞിരിക്കുന്നു.

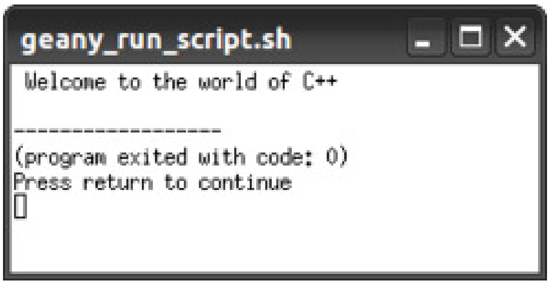
## D. പ്രോഗ്രാം റൺ/എക്സിക്യൂട്ട് ചെയ്യുക.

ഒരു കമ്പ്യൂട്ടർ പ്രോഗ്രാമിലെ നിർദേശങ്ങൾ കമ്പ്യൂട്ടർ പ്രാവർത്തികമാക്കുന്ന പ്രക്രിയയാണ് പ്രോഗ്രാം റണ്ണിംഗ്. പ്രോഗ്രാം റൺ ചെയ്യാനായി ബിൽഡ് മെനുവിലെ എക്സിക്യൂട്ട് ഓപ്ഷൻ

തിരഞ്ഞെടുക്കുക. ടൂൾ ബാറിലുള്ള എക്സിക്യൂട്ട് ബട്ടൺ ക്ലിക്ക് ചെയ്തുകൊണ്ടും പ്രോഗ്രാം എക്സിക്യൂട്ട് ചെയ്യാവുന്നതാണ്. ഔട്ട്പുട്ട് ചിത്രം 5.4-ൽ കാണുന്നത് പോലെ ഒരു പുതിയ വിൻഡോയിൽ പ്രദർശിപ്പിക്കുന്നതാണ്.

## E. IDE യുടെ പ്രവർത്തനം അവസാനിപ്പിക്കുക

ഒരിക്കൽ നാം പ്രോഗ്രാം എക്സിക്യൂട്ട് ചെയ്യുകയും പ്രതീക്ഷിച്ച ഔട്ട്പുട്ട് ലഭിക്കുകയും ചെയ്ത് കഴിഞ്ഞാൽ, ഫയൽ മെനുവിൽ നിന്ന് ക്ലോസ് ഓപ്ഷൻ തിരഞ്ഞെടുത്തോ സജീവമായ ടാബിലെ ക്ലോസ് ബട്ടൺ (X) ക്ലിക്ക് ചെയ്തോ പ്രോഗ്രാം അവസാനിപ്പിക്കാം. പ്രോഗ്രാമിങ് തുടരുന്നുവെങ്കിൽ മുമ്പ് വിവരിച്ചത് പോലെ പുതിയ ഫയൽ തുറന്ന് അതിൽ പ്രോഗ്രാം ടൈപ്പ് ചെയ്യാവുന്നതാണ്. അല്ലെങ്കിൽ ടൈറ്റിൽ ബാറിലെ ക്ലോസ് ബട്ടൺ (X) ക്ലിക്ക് ചെയ്ത് IDE- യുടെ പ്രവർത്തനം അവസാനിപ്പിക്കാം. ഇതിനു പുറമേ Ctrl+Q കീ-കൾ ഒരുമിച്ചു ഉപയോഗിച്ചോ ഫയൽ മെനുവിൽ നിന്ന് ക്വിറ്റ് ഓപ്ഷൻ തിരഞ്ഞെടുത്തോ IDE-ൽ നിന്ന് പുറത്തുവരാം.

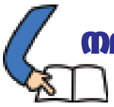



ചിത്രം 5.4: ഔട്ട് പുട്ട് വിൻഡോ

ഉബുണ്ടു ലിനക്സിലെ ജിനി IDE-യിൽ .cpp എന്ന എക്സ്റ്റൻഷൻ നൽകി സേവ് ചെയ്ത ഒരു സോഴ്സ് പ്രോഗ്രാം കമ്പയിൽ ചെയ്യുമ്പോൾ ലഭിക്കുന്ന ഫയലിന്റെ എക്സ്റ്റൻഷൻ .o (ഒബ്ജക്ട് ഫയൽ) എന്നായിരിക്കും. പ്രോഗ്രാം ലിങ്ക് ചെയ്യുമ്പോൾ ഒബ്ജക്ട് ഫയലിൽ നിന്ന് പ്രൊസസ്സറിന് എക്സിക്യൂട്ട് ചെയ്യാൻ വേണ്ട .out എന്ന എക്സ്റ്റൻഷനോട് കൂടിയ എക്സിക്യൂട്ടബിൾ ഫയൽ ലഭിക്കും. ഈ ഫയലാണ് പ്രൊസസ്സർ എക്സിക്യൂട്ട് ചെയ്യുന്നത്.

**നമുക്ക് ചെയ്യാം**

1. *"SMOKING IS INJURIOUS TO HEALTH" എന്ന സന്ദേശം സ്ക്രീനിൽ പ്രദർശിപ്പിക്കാനായി ഒരു പ്രോഗ്രാം എഴുതുക.*
2. *"TOBOCCO CAUSES CANCER" എന്ന സന്ദേശം മോണിട്ടറിൽ പ്രദർശിപ്പിക്കാനായി ഒരു പ്രോഗ്രാം എഴുതുക*



## നമുക്ക് സംഗ്രഹിക്കാം

1980-കളുടെ തുടക്കത്തിൽ Bjarne Stroustrup ആണ് C++ വികസിപ്പിച്ചത്. C++-ന് അതിന്റേതായ ക്യാരക്ടർ സെറ്റുണ്ട്. ടോക്കൺ എന്നത് പ്രോഗ്രാമിന്റെ ഏറ്റവും ചെറിയ ഘടകവും അവ നിർമ്മിക്കപ്പെട്ടിരിക്കുന്നത് C++-ലെ ഒന്നോ അതിലധികമോ അക്ഷരങ്ങൾ ചേർന്നുമാണ്. കീവേഡുകൾ, ഐഡന്റിഫയറുകൾ, ലിറ്ററലുകൾ, പഞ്ചുവേറ്ററുകൾ, ഓപ്പറേറ്ററുകൾ എന്നിങ്ങനെയുള്ള അഞ്ച് തരം ടോക്കണുകളുണ്ട്. പ്രോഗ്രാമുകൾ കമ്പ്യൂട്ടറിൽ എഴുതുന്നത് ഒരു എഡിറ്ററിന്റെ സഹായത്താലാണ്. സോഴ്സ് കോഡ് കമ്പ്യൂട്ടറിൽ നൽകുന്നതിനും അത് കമ്പയിൽ ചെയ്യുന്നതിനും ഒബ്ജക്ട് കോഡ് എക്സിക്യൂട്ട് ചെയ്യുന്നതിനും വേണ്ടി GCC-യും ജിനി IDE-യും പോലുള്ള സോഫ്റ്റ്വെയറുകൾ സൗകര്യമൊരുക്കുന്നു.

## പഠന നേട്ടങ്ങൾ

ഈ അധ്യായത്തിന്റെ പൂർത്തീകരണത്തിന് ശേഷം പഠിതാവിന് കഴിയും.

- C++-ലെ ക്യാരക്ടർ സെറ്റ് ലിസ്റ്റ് ചെയ്യുക.
- വിവിധ ടോക്കണുകളെ തരം തിരിക്കുക.
- കീവേഡുകളെ തിരിച്ചറിയുക.
- സാധുവായ ഐഡന്റിഫയറുകൾ എഴുതുക.
- ലിറ്ററലുകൾ തരം തിരിക്കുക.
- ജിനി IDE-യുടെ പ്രധാന ഭാഗങ്ങൾ തിരിച്ചറിയുക.
- ഒരു ലളിതമായ പ്രോഗ്രാം എഴുതി കമ്പയിൽ ചെയ്ത് റൺ ചെയ്യുക.

## മാതൃകാ ചോദ്യങ്ങൾ

### വളരെ ലഘുവായ ഉത്തരങ്ങൾ എഴുതുന്നവ.

1. C++ ക്യാരക്ടർ സെറ്റിലെ വിവിധ ക്യാരക്ടറുകൾ എന്തെല്ലാം?
2. എന്താണ് എസ്കേപ്പ് സീക്വൻസ്?
3. ആരാണ് C++ വികസിപ്പിച്ചത്?
4. ടോക്കണുകൾ എന്നാലെന്ത്? C++-ലെ ടോക്കണുകളുടെ പേരെഴുതുക.
5. എന്താണ് C++-ലെ ഒരു ക്യാരക്ടർ സ്ഥിരാങ്കം?
6. എങ്ങനെയാണ് ചിത്രീകരിക്കാനാകാത്ത ക്യാരക്ടറുകൾ C++-ൽ പ്രതിനിധാനം ചെയ്യുന്നത്? ഉദാഹരണം നൽകുക.
7. \(സ്ലാഷ്), ‘ (ഏക ഉദ്ധരണി), “ (ജോഡിയായ ഉദ്ധരണി), ? (ചോദ്യച്ചിഹ്നം) തുടങ്ങിയ ക്യാരക്ടറുകൾ എസ്കേപ്പ് സീക്വൻസുകൾ ഉപയോഗിച്ച് ടൈപ്പ് ചെയ്യുന്നത് എന്ത് കൊണ്ടാണ്?
8. ഏത് എസ്കേപ്പ് സീക്വൻസുകളാണ് ന്യൂലൈൻ ക്യാരക്ടറിനേയും നൾ ക്യാരക്ടറിനെയും പ്രതിനിധാനം ചെയ്യാൻ ഉപയോഗിക്കുന്നത്?
9. ഒരു എസ്കേപ്പ് സീക്വൻസ് ..................... ക്യാരക്ടറുകളെ പ്രതിനിധാനം ചെയ്യുന്നു.
10. താഴെ തന്നിട്ടുള്ളവയിൽ ഏതൊക്കെയാണ് C++-ലെ സാധുവായ ക്യാരക്ടറുകൾ/സ്ട്രിങ് സ്ഥിരാംഗങ്ങൾ?

    ‘c’ ‘anu’ “anu” mine ‘min’s’ “ ”

    ‘char ‘\ \’

11. എന്താണ് ഫ്ളോട്ടിങ് പോയിന്റ് സ്ഥിരാംഗം? അവ പ്രതിനിധാനം ചെയ്യുന്നതിനുള്ള വിവിധ മാർഗങ്ങൾ എന്തെല്ലാം?
12. എന്താണ് C++-ലെ സ്ട്രിങ് ലിറ്ററലുകൾ? ക്യാരക്ടർ സ്ഥിരാംഗങ്ങളും സ്ട്രിങ് ലിറ്ററലുകളും തമ്മിലുള്ള വ്യത്യാസമെന്ത്?
13. റൺ ചെയ്യുവാനുപയോഗിക്കുന്ന ഒരു C++ ഫയലിന്റെ എക്സ്റ്റൻഷൻ എന്താണ്?
14. താഴെ തന്നിട്ടുള്ളവയിൽ നിന്ന് അസാധുവായ ഐഡന്റിഫയറുകൾ കണ്ടെത്തുക. അസാധുവായതിന് കാരണമെഴുതുക.

    (a) Principal amount (b) Continue (c) Area (d) Date-of-join (e) 9B
15. ലേബൽ എന്നത് C++-ലെ ഒരു .................ആണ്

    (a) കീവേർഡ് (b) ഐഡന്റിഫയർ (c) ഓപ്പറേറ്റർ (d) ഫങ്ഷൻ
16. താഴെത്തന്നിട്ടുള്ള ടോക്കണുകൾ C++ പ്രോഗ്രാമിൽ നിന്ന് എടുത്തിട്ടുള്ളവയാണ്. താഴെ തന്നിട്ടുള്ള പട്ടികയിൽ അവ യഥാസ്ഥാനത്ത് നിരത്തുക.

    (int, cin, %, do, =, “break”, 25.7, digit)

| കീവേഡുകൾ | ഐഡന്റിഫയറുകൾ | ലിറ്ററലുകൾ | ഓപ്പറേറ്ററുകൾ |
|---|---|---|---|
| | | | |

**ലഘു ഉത്തരങ്ങളെഴുതുന്നവ.**

1. ഐഡന്റിഫയറുകളെ പരിപാലിക്കുന്ന നിയമങ്ങളെഴുതുക.
2. എന്താണ് C++-ലെ ടോക്കണുകൾ? എത്ര തരം ടോക്കണുകൾ C++-ൽ അനുവദിക്കുന്നു? അവ ലിസ്റ്റ് ചെയ്യുക.
3. കീവേഡുകളും, ഐഡന്റിഫയറുകളും വേർതിരിക്കുക
4. പൂർണ സംഖ്യാ സ്ഥിരാംഗങ്ങളെ C++-ൽ എങ്ങനെ പ്രതിനിധാനം ചെയ്യുന്നു? ഉദാഹരണ സഹിതം വിശദീകരിക്കുക
5. C++-ലെ ക്യാരക്ടർ സ്ഥിരാംഗങ്ങൾ എന്തെല്ലാം? എങ്ങനെ അവ പ്രയോഗത്തിൽ വരുത്തുന്നു?

**വിവരണാത്മക ഉത്തരങ്ങളെഴുതുന്നവ.**

1. വിവിധയിനം ടോക്കണുകളെക്കുറിച്ച് ചുരുക്കി വിവരിക്കുക.
2. വിവിധയിനം ലിറ്ററലുകളെക്കുറിച്ച് ഉദാഹരണ സഹിതം വിശദീകരിക്കുക.
3. ജിനി IDE-യെക്കുറിച്ചും അതിന്റെ പ്രത്യേകതകളെക്കുറിച്ചും ചുരുക്കി വിശദീകരിക്കുക.