

C++ ပရိုဂရမ်ကို ဘယ်လို စရေးမယ်ဆိုတာ ပြောပြမှာ ဖြစ်ပါတယ်။ပရိုဂရမ် ဆိုတာက အစီအစဉ်တွေရေးတာပဲ ဖြစ်ပါတယ်။ဥပမာ ထမင်းချက်မယ်ဆိုရင် ပထမဆုံးဆန်ဆေးရမယ်။ နောက်အိုးထဲ ထည့်ရမယ်။

နောက်ရေခမ်းချက်မှာလား...

ရေငဲ့ချက်မှာလား စဉ်းစားရတော့မယ်။ အဲ့လိုစဉ်းစားတဲ့အချိန်မှာ ထွက်လာတဲ့အဖြေက ရေခမ်းဆိုရင်

ရေဘယ်လောက် ထည့်ရမယ်...တကယ်လို့ရေခမ်းမဟုတ်ဘူးဆိုရင် ရေဘယ်လောက်

ထည့်ရမယ်။Programming မှာဆိုရင် .. ရေခန်းချက်မှာလား...

ရေငဲ့ချက်မှာလားဆိုတဲ့အပိုင်းက condition ပိုင်းပေါ့ ။

ပရိုဂရမ်ဆိုတာ အစီအစဉ်ကျကျလုပ် ရပါတယ်။ ပထမဆုံး လုပ်ရမယ့်ဟာကိုပထမဆုံးလုပ်တယ်။ ဆိုလိုတာက ရေခမ်းလား...ရေငဲ့လားစဉ်းစားရမယ့်ဟာကို ထမင်းအိုး မီးဖိုပေါ်ရောက်သွားမှ စဉ်းစားလို့ မရတော့ဘူး။ အဲ့လိုရေခမ်းလား...ရေငဲ့လားလို့ စဉ်းစားလိုက်တာနဲ့တင် ပရိုဂရမ်က ၂ ခုကွဲသွားပြီရေခန်းဆိုရင် command တစ်မျိုး ၊ ရေငဲ့ ဆိုရင် command တစ်မျိုးပေါ့။ ဒါပေမယ့် နောက်ဆုံးထွက်လာတဲ့ အဖြေမှာတော့ သွား တူမယ်။အခု ပြောနေတာက Algorithm လို့ခေါ်တဲ့ programming ရဲ့ အခြေခံအကျဆုံး language တစ်မျိုးဖြစ်ပါတယ်။ ပရိုဂရမ်တစ်ခု စရေးခင် မှာစဉ်းစားတဲ့အပိုင်းဖြစ်ပါတယ်။ flow chart ပုံမှာ မီးချောင်းမလင်းဘူးပေါ့ ။ အဲ့ချိန်မှာ ပထမဆုံး စစဉ်းစားတာက ပလပ်တပ်ထားလားပေါ့။ No ဆိုရင် plug in lamp ကိုသွားမယ် ။ Yes ဆိုရင် နောက်တစ်ဆင့်ထပ်စစ်ဖို့လိုလာပါပြီ။

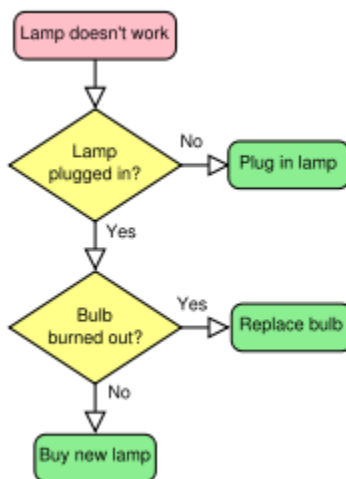


image taken from <http://en.wikipedia.org/wiki/Flowchart>

ဂဏန်း၂ လုံး ဖြစ်တဲ့ ၁ နဲ့ ၂ ကို ပေါင်းရင် ဦးနှောက်ထဲမှာ ဘယ်လိုစဉ်းစားပါသလဲ။ ၁ နဲ့ ၂ ပေါင်းဘယ်လောက်ရလဲ ဆိုတော့ ၃ ရတယ်ဆိုတာ ချက်ချင်းပဲ ဦးနှောက်ကနေ အဖြေထွက်ပါတယ်။ ဒါဆို ၁၂၃၄၅ ၃၃၃ နဲ့ ၃၂၉၀၀၉၂ ပေါင်းရင်ရော...။ဒီတစ်ခါကျတော့ ဦးနှောက်ထဲမှာ တွက်လို့ မမြန်တော့ပါဘူး ။ အဲဒီ ဂဏန်း ၂ ခုကိုပေါင်းတော့မယ်ဆိုရင် ၁၂၃၄၅ ၃၃၃ ဂဏန်းကို အရင်စိတ်ထဲမှာမှတ်လိုက်ရမယ်။ ပြီးမှ ၃၂၉၀၀၉၂ ဆိုတာကို ပေါင်းထည့်လိုက်မယ်။ နောက်ဆုံးမှာ အဖြေ .. ဆိုတာ ထွက်လာမယ်။ ဆိုလိုတာကအလုပ် ၃ ခုလုပ်ရတယ်။ ပထမ ဂဏန်း ကိုမှတ်ရတယ်။ ဒုတိယဂဏန်းကိုထပ်မှတ်ပြီး ပထမဂဏန်းနဲ့ပေါင်းရတယ်။ နောက်ဆုံးမှာ အဖြေဆိုတာကိုထပ်မှတ်ရတယ်။ အဲ့ဒီမှာလဲ အစီအစဉ် ကျနု်ုဆိုတာရှိတယ်။ ပေါင်းတာကိုအရင်လုပ်ပြီးမှ ဂဏန်းမှတ်လို့ မရပါဘူး။ ဂဏန်းတွေအရင်မှတ်ပြီးမှပေါင်း ရပါတယ်။ ပေါင်းပြီးမှ အဖြေဆိုတာ ကိုထပ်မှတ်ရပါတယ်။

programming language တိုင်းမှာ .. machine language ကိုပြောင်းပေးဖို့အတွက် compiler

ဆိုတာလိုပါ တယ်။ machine language ဆိုတာ .. 101010 ကိုပြောတာပါ။ ကွန်ပျူတာ ဆိုတာ ဆားကစ် တွေနဲ့ တည်ဆောက်ထားတဲ့အတွက်ကြောင့်၁၀၀၁၀၀၁ ကိုပဲနားလည်ပါတယ်။ ဥပမာဆိုပါတော့ ..မြန်မာယူနီကုတ်ကို opera တို့ တစ်ခြား ဂိုဂယ် ဘရောက်ဇာတို့မှာ မမြင်ရဘူး။ ဘာကြောင့်လဲဆိုတော့ အဲဒီ software တွေမှာ မြန်မာယူနီကုတ်ကို ပြောင်းပေးနိုင်တဲ့ ပရိုဂရမ်မပါသေးတဲ့အတွက် ဖြစ်ပါတယ်။ အဲ့လိုပဲ php ကိုစက်ထဲမှာ run ကြည့်မယ်ဆိုရင်လဲ သူ့ကိုsupport ပေးတဲ့ဆော့ဝဲတစ်ခုတင်ပေးရပါတယ်။ ဆိုလိုတာက c++ လဲအဲ့လိုပဲ c++ ရေးဖို့အတွက် compiler တစ်ခုလိုပါတယ်။

ခုနက ဂဏန်း၂ လုံးပေါင်းတာဆက်ပြောရမယ် ဆိုရင်၁ နဲ့ ၂ ကိုပေါင်းရတာ စဉ်းစားရတာလွယ်တယ် ။ ဒါပေမယ့် အရမ်းများတဲ့ ဂဏန်း၂ လုံးဆိုရင်ကျမလွယ်တော့ဘူး။ ကွန်ပျူတာရဲ့ process လုပ်တဲ့ အပိုင်းက လူထက် အများကြီး ပိုမြန်တယ်ပေါ့။ အဲဒီတော့ လူကနေပြီး ကွန်ပျူတာကို ဂဏန်း၂ လုံး ပေါင်းနည်း သင်ပေးလိုက်တာ ။ ငါပေါင်းရင်ဒီလိုပေါင်းတယ်.... မင်းလဲ ဒီလိုပေါင်းဆိုပြီး။

ပထမဆုံး .. ဂဏန်းတစ်လုံးကိုမှတ်ပါ..နောက် ဂဏန်းတစ်လုံး ကိုထပ်မှတ်ပါ။ နောက် အဲ့ဒီ၂ ခုကိုပေါင်းထည့်လိုက်ပါ။ နောက်အဖြေ ထုတ် ပြပါဆိုပြီး....။ အဲ့ဒါကြောင့် programming လို့ခေါ်ခြင်း ဖြစ်ပါတယ်။ အစီအစဉ်တွေရေးပေးရုံပါပဲ။

ဂဏန်း၂ လုံးပေါင်းတဲ့ပရိုဂရမ်တစ်ခုနဲ့ ဖြစ်တဲ့ ဒီပရိုဂရမ်လေးကို ကြည့်ပေးပါ။

```
#include
int a=10;
int b=20;
int c;
c=a+b;
cout << c;
```

၁၀ နဲ့ ၂၀ ပေါင်းတဲ့ပရိုဂရမ် ဖြစ်ပါတယ်။ #include ဆိုတာက ပေါင်းနှုတ်မြှောက်စားတွေအတွက် အလုပ်လုပ်ပေးတဲ့ ဖိုင်ကိုခေါ်သုံးတာ ဖြစ်ပါတယ်။

int a=10; ဆိုတာက int က ကိန်းပြည့် (integer) ကို ပြောတာ ဖြစ်ပါတယ်။
a ဆိုတဲ့ဟာကို တန်ဖိုး၁၀ လို့သတ်မှတ်ပေးလိုက်တာ။

int b=20; လဲ အဲဒီအတိုင်း အတူတူပဲ ဖြစ်ပါတယ်။

c=a+b; အဲ့ဒါက c ထဲကို အေနှု.ဘီပေါင်းထည့်လိုက်ခြင်း ဖြစ်ပါတယ်။ cout << c; ဆိုတာက cout << ကထုတ်ပြတဲ့ command ဖြစ်ပါတယ်။
c; တန်ဖိုးကိုထုတ်ပြမှာ ဖြစ်ပါတယ်။

အင်ဂျင်နီယာတစ်ယောက်က အိမ်ဆောက်တဲ့ပရိုဂရမ်ရေးမယ် လို့စဉ်းစားကြည့်လိုက်ရအောင်။ ဥပမာ .. တိုက်ရဲ့အမြင့်တွေ ၊ ဧရိယာတွေ ရိုက်ထည့်လိုက်တာနဲ့ ဘီလပ်မြေဘယ်လောက်ကုန်မယ်ပေါ့။ တကယ်တမ်း သေချာစဉ်းစားကြည့်ရင် လက်နဲ့ချတွက်တာမှမြန်ဦးမယ်။ ရေးတာကကြာမှာပေါ့ ။ ဒါပေမယ့် အကြိမ်ပေါင်းများစွာအတွက် ၊ ဂဏန်း ပေါင်းများစွာအတွက်ဆိုရင်ကျတော့ စာရွက်ထဲမှာ ချတွက်ဖို့ မလွယ်ကူတော့ပါဘူး။ အဲဒီအတွက် programming ဟာ လိုအပ်လာခြင်း ဖြစ်ပါတယ်။

သက်တန်း

```
/* Font Definitions */
@font-face
{font-family:Zawgyi-One;
panose-1:2 11 6 4 3 5 4 4 2 4;
mso-font-charset:0;
mso-generic-font-family:swiss;
mso-font-pitch:variable;
mso-font-signature:1627421319 -2147483648 8 0 66047 0;}
/* Style Definitions */
p.MsoNormal, li.MsoNormal, div.MsoNormal
{mso-style-parent:"";
margin:0cm;
margin-bottom:.0001pt;
mso-pagination:widow-orphan;
font-size:12.0pt;
font-family:"Times New Roman";
mso-fareast-font-family:"Times New Roman";}
@page Section1
{size:612.0pt 792.0pt;
margin:72.0pt 90.0pt 72.0pt 90.0pt;
mso-header-margin:35.4pt;
mso-footer-margin:35.4pt;
mso-paper-source:0;}
div.Section1
{page:Section1;}
-->
```

(၂၀၀၈
ခုနှစ် ၊ သြဂုတ်လ ၂၈ ရက်က ဖော်ပြခဲ့ပြီးသော ဆောင်းပါးဖြစ်ပါသည်။

မူလဆောင်းပါးတွင်
ကွန်မန်များကို ဖွင့်ပေးခဲ့စဉ်က ကြော်ငြာ spam ကွန်မန်များ ၆၅၀၅ ခု ဝင်ရောက်နေသဖြင့်
ဖျက်ပယ်လိုက်ပြီးနောက် ယခု အသစ်တဖန် ပြန်လည် တင်ပြခြင်းဖြစ်ပါသည်။)

C++ လေ့လာကြမယ် (အပိုင်း- ၂)

```
// a small C++ program
#include
```

```
int main()
{
    std::cout << "Hello, world!" << std::endl;
```

```
    return 0;
}
```

ပရိုဂရမ်အသေးလေးဖြစ်တဲ့ Hello, World! ဆိုတဲ့ ပရိုဂရမ်လေးနဲ့ စလိုက်ကြရအောင်ပါ။

Hello, world! ဆိုတဲ့ စာတမ်းလေး ထုတ်ပြတဲ့ ပရိုဂရမ် ဖြစ်ပါတယ်။ programming မီးတိုင်းမှာ space ကို တစ်ခုထက်ပိုပြီးနားမလည်ပါဘူး။
// a small C++ program// နဲ့ အမြဲအဲဒါနဲ့ စရပါတယ်။

// စပေ့ ပြီးရင် ကိုယ်ရေးချင်တာရေးလို့ရပါတယ်။ ပရိုဂရမ် run တဲ့အထဲမှာ အဲဒါကို ထည့်မ run ပါဘူး။ ကုတ်တွေနဲ့ ကိုယ်ရေးတာနဲ့ကို အမြဲ စပေ့ခြား ပေးရပါတယ်။ // a small C++ program ဆိုတာက ထည့်ရင်လဲရတယ် မထည့်ရင် လဲရပါတယ်။ မထည့်လဲ Hello, World! လို့ထွက်မှာပါ။ ထည့်လဲ Hello, World!

လိုထွက်မှာပါ။ ဘာလို့ထည့်တယ်ဆိုတာက ဒီလိုပါ။ ပရိုဂရမ်အသေးတွေအတွက်တော့ အဲဒါဟာအသုံးမဝင်ပါဘူး။ ဒါပေမယ့် လိုင်းပေါင်း မြောက်များစွာ လိုင်း ၁ ထောင် ၊ လိုင်း ၂ ထောင် စသည်ဖြင့် function ဖန်ရှင်ပေါင်းများစွာရေးလာတဲ့အချိန်မှာ ပရိုဂရမ်တစ်ပုဒ်က အခုတစ်ခါတည်းရေးလိုက်တယ် အခုတစ်ခါတည်း အဆင်ပြေသွားတယ်လို့မရှိပါဘူး။ အမှားတွေလိုက်ပြင်ရတယ်။ စာတွေကျခဲ့တာတို့ ကွန်မန်တွေ မှာ မှားသွားတာတို့၊ ; ဆိုတာလေး ကျခဲ့တာတို့၊ အဲဒါမျိုးတွေ ဖြစ်ခဲ့ရင် မှတ်စုအနေနဲ့ မှတ်ထားတဲ့သဘောပါ။

သူ့ရဲ့လုပ်ပိုင်ခွင့်ရှိတဲ့ ဖန်ရှင်ကတော့ // နောက်မှာကြိုက်တာရေး .. ဒါကို run တဲ့အချိန်မှာထည့်မ run ပါဘူးတဲ့။ အဲဒါသူ့ရဲ့လုပ်ပိုင်ခွင့်ပါ။ programming က သူ့ကုတ်လေးတွေနဲ့ သူ လုပ်ပိုင်ခွင့်ပေးထားတယ်။ အဲဒီလုပ်ပိုင်ခွင့်လေးတွေကို .. ကိုယ်က ကိုယ်လိုချင်တဲ့ပုံစံဖြစ်အောင် သုံးယူရတာပါ။

အခုတစ်ခါ #include ဆိုတာလာပါပြီ။ ဒါက ဒုတိယအကြောင်း ကိုရှင်းလင်းချက်ပေါ့။ #include အဲဒါကဘာလဲဆိုရင် c++ ကိုရေးမယ့် ကွန်ပိုင်လာထဲမှာ library ဆိုတာရှိတယ်။ ပေါင်း နှုတ် မြောက် စားဆိုရင်လဲ ပေါင်းနှုတ်မြောက်စားဖိုင် ၊ input output input output ဖိုင်....iostream ဆိုတာ အဲဒီ library ထဲက input output ကိုအလုပ်လုပ်ပေးတဲ့ဖိုင်ပါ။ #include မှာ #include <သုံးမယ့်ဖိုင်နာမည်> အဲလိုရေးရပါတယ်။ ဒီပရိုဂရမ်လေးမှာတော့ #include တစ်ခုပဲခေါ် သုံးထားပါတယ်။ ဘာလို့လဲဆိုတော့ သူက Hello, World! ဆိုတာလေးပဲပါတာကြောင့်ပါ။

Hello, World! ဆိုတဲ့output လေးပဲပါတော့input output ဖိုင်လေးပဲယူသုံးတာဖြစ်ပါတယ်။ တကယ်လို့ input output လဲပါမယ် ..ပေါင်းနှုတ်မြောက်စားတွေလဲပါတော့မယ်ဆိုရင် #include #include အဲလို ၂ခုဖြစ်သွားမှာပါ။ ကိုယ် လုပ်မယ့်အလုပ်ပေါ်မှာမူတည်ပြီး ၂ ခု၊ ၃ ခု ၊ ၄ ခု စသည်ဖြင့်

ကွဲသွားပါတယ်။

www.teachtutorials.com ကွန်ပျူတာ နည်းပညာဆိုင်ရာ Tutorial

များကိုအခမဲ့လေ့လာနိုင်တဲ့ဝက်ဘ်ဆိုက်တစ်ခုဖြစ်ပါတယ်။ ဝက်ဘ်ဒီဇိုင်းသင်ကြားမှုများ ၊

Unix Tutorial များ ၊Flash Tutorial

များအစရှိသဖြင့်ခေါင်းစဉ်ဆယ်မျိုးခွဲခြားသက်မှတ်ထားပါတယ်။ ဝက်ဘ်ဆိုက်အတွင်း Tutorial

ပေါင်း၅၀၀၀ ဝန်းကျင်ထည့်သွင်းပေးထားပြီးအလွယ်တကူ အခမဲ့လေ့လာနိုင်ပါတယ် ။

Register ပြုလုပ်ပေးရမှာဖြစ်ပါတယ်။

www.pemdirveapps.com