Note Title 27-08-2018

COL 100

Aug 27, 2018

```
int j=10;
for( int i=0; i<j; j--){
    int j = i++;
    cout << j << " ";
}
```

1 9
2 8
3 7
4 6
j=i;
i=i+1;

0 1 2 3 4

```
int a=1,b=1, c;
cout<<1<<2;
cout<<12;
```

```
int x=5
    x= x+5 ;
```

x= x+5;

⇒ x += 5;

x op= 5;

+, -, *, /, %

x -= 5;

⇓

x = x op 5;

```
if (x>5){
    x=6; }
} else{
    x=7;
}
```

```
for ( int i=0; i<5; i++)
        stmt;
→ stmt 2;
```

Signature

void printGreetings(string name, int cnt){
}

Functions:-

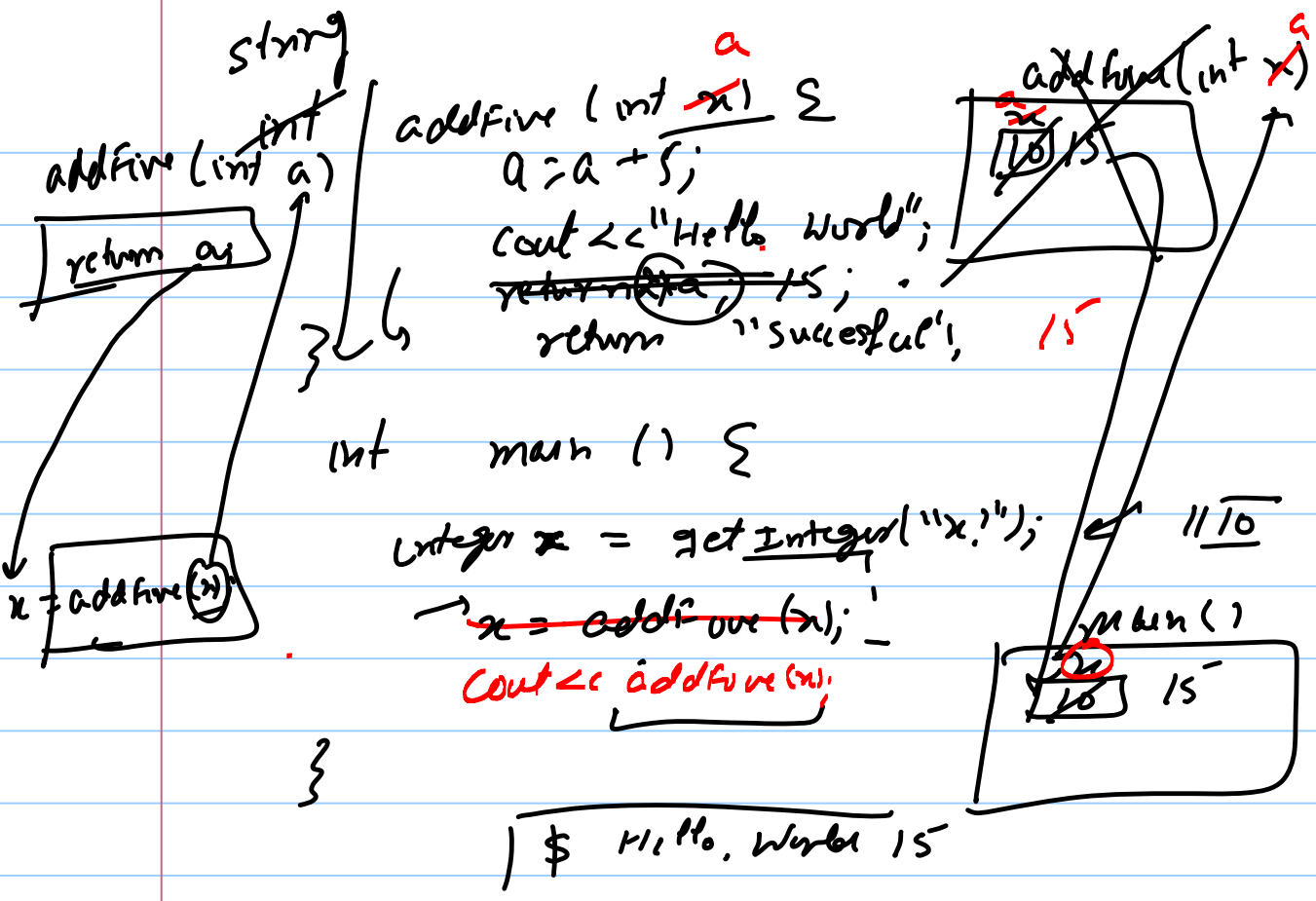
Typer function_name(type1 arg1, ... typen argn){

stmt1;
stmt2;
:
stmtk;
return expr;

function declaration

Function Body }

int main(){
Typer var=function_name(arg1, ... , argn);
}


string
int
addFive (int a)
return a;
x = addFive (x);
addFive (int x) {
a = a + 5;
cout << "Hello World";
return "successful";
}
int main () {
Integer x = getInteger("x:");
x = addFive (x);
cout << addFive (x);
}
addFive (int x)
a
x
10
15
main ()
x
10
15
$ Hello. World 15

→ int addFive(int x); → Function declaration

↓ [pass by value] "copies" the arguments

```
int main() {
    int x = getInteger("x?");
    x = addFive(x);
    cout << x;
}

int addFive(int x) {
    x = x+5;
    return x;
}
```

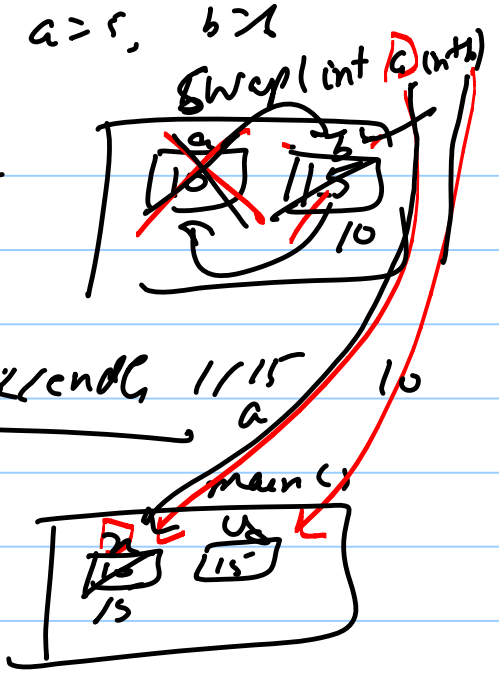
→ function body

Pass by reference

```
void swap (int &a, int &b) {
    int tmp = a;
    a = b;
    b = tmp;
    cout << a << " " << b << endl;   // 15 10
}

int main () {
    int x = 10;
    int y = 15;
    swap(x, y);
    cout << x << " " << y << endl;   // 15 10
}
```

a = 5, b = 1



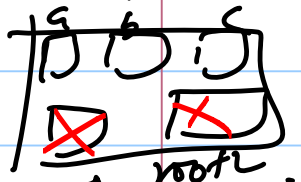
$f(x) = ax^2 + bx + c$

$d = \sqrt{b^2 - 4ac}$

$root1 = \frac{-b + d}{2a}$

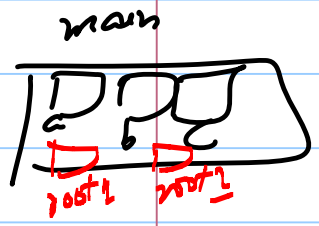
$root2 = \frac{-b - d}{2a}$

```
#include <math>
```




```
void getQuadraticRoots(double a, double b, double c,
                       double &root1, double &root2) {
    double d = sqrt(b*b - 4*a*c);
    root1 = (-b + d)/(2*a);
    root2 = (-b - d)/(2*a);
}

int main() {
    double a = getReal("a?");
    double b = getReal("b?");
    double c = getReal("c?");
    getQuadraticRoots(a, b, c, root1, root2);
}
```



```
int main() {
    double a = getReal("a?");
    double b = getReal("b?");
    double c = getReal("c?");
    double root1, root2;   r1, r2;
    getQuadraticRoots(a, b, c, root1, root2);
                                r2     r2
}
```

Strings

sequence of characters

```
#include <string>
#include <iostream>
int main() {
    string s = "COL 100 !!";
    cout << s;
    s = "Manav was not so great";
    cout << s << endl;
}
```

s[0] s[1] s[2] -- s[7]

indexed from 0

reality

s[0] - - s[21]

s[0] = 'R';