

Day2_Intro To Programming

Day2 Content:

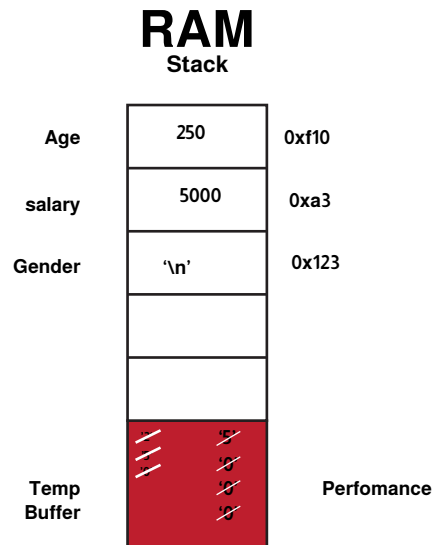
- Buffer
- Operators
- Control Flow
 - Conditional Statment
 - Iterations

```

1 int main(){
2     int age;
3     float salary;
4     char gender;
5     printf("Please Enter Your Age\n");
6     scanf("%i", &age);
7     printf("Please Enter Your Salary\n");
8     scanf("%f",&salary);
9     printf("Please Enter Your Gender\n");
10    scanf("%c",&gender);
11
12
13    printf("\n_____ \n");
14    printf("Your Age Is %i\n", age);
15    printf("Your Salary Is %f\n", salary);
16    printf("Your Gender Is %c\n", gender);
17    printf("\n_____ \n");
18    return 0; //End
19 }
20

```

Please Enter Your Age
250
Please Enter Your Salary
5000
Please Enter Your Gender



Operators

- Arthematical operators
- Assignment Operators
- Relational/Comparision Operators
- Logical Operators
- Bitwise Operators

Unary

- Increment
- Decrement

Binary

- Arthematical operators
- Assignment Operators
- Relational/Comparision Operators
- Logical Operators

Tirmary
()? : Search Ya Bashmohandess

1

2

Arthematical operators

لا تؤثر فى قيمة المتغير فى الذاكرة

- +
-
- /
- *
- %

```

1 printf("%i + %i = %i\n", Val1, Val2, Val1 + Val2); //15
2 printf("%i - %i = %i\n", Val1, Val2, Val1 - Val2); //5
3 printf("%i / %i = %i\n", Val1, Val2, Val1 / Val2); //2
4 printf("%i * %i = %i\n", Val1, Val2, Val1 * Val2); //50

```

```

1 int x = 10;
2 int y = 20;
3 int sum = x + y;

```

x	10
y	20
sum	30

لاحظ القيم هنا لا تتأثر ب Arth Op

$$10 \% 2 = 0$$

$$11 \% 2 = 1$$

$$14 \% 3 = ?$$

$$\begin{array}{r}
 * \quad 4 \\
 3 \overline{) 14} \\
 \underline{12} \\
 2
 \end{array}$$

4

$$27 \% 5 = ??$$

$$\begin{array}{r}
 * \quad 6 \\
 4 \overline{) 27} \\
 \underline{24} \\
 3
 \end{array}$$

3

باقى القسمة ليها 5 حالات

1. فى حالة باقى قسمة أ % ب وكان الناتج = 0 -- يعني أ تقبل القسمة على ب، فى حالة != 0 -- يعني أ لا تقبل القسمة على ب
2. فى حالة باقى قسمة أ % 2 وكان الناتج = 0 -- معناه ان أ هى عدد زوجى واذا كان 1 معناه انه فردى
3. فى حالة باقى قسمة أ % ب دائما الناتج يكون اصغر من المقسوم عليه
4. فى حالة باقى قسمة أ % ب وكانت قيمة أ اصغر من ب -- دائما الناتج هو قيمة أ
5. فى حالة باقى قسمة أ % 10 دائما النتيجة هى الاحاد

5

Assignment Operators

تؤثر فى قيمة المتغير فى الذاكرة

=
+=
-=
*=
/=

```
1 int x = 10;
2 x += 10; //20;
3 x -= 5; //15
4 x *= 5; //75
5 x /= 5; //15
6 x %= 5; //0
7 //-----
8 x = x + 10; //20;
9 x = x - 5; //15
10 x = x * 5; //75
11 x = x / 5; //15
12 x = x % 5; //0
```

int x = 10; x 20

x += 10; ;

```
1 int x=10;
2 //Postfix
3 printf("%i\n",x++); //10
4 printf("%i\n",x); //11
```

10
11

- 1- Print Value Of X
- 2- Increment x X in Mem

Increment

Postfix++
++Prefix

Decrement

Postfix--
--Prefix

x 11 0x10

```
1 int x=10;
2 ++x; //Value=11
3 printf("%i\n",x++); // Print=11 //Value=12
4 printf("%i\n",++x); // Print=13 //Value=13
5 printf("%i\n",++x); // Print=14 //Value=14
6 printf("%i\n",x++); // Print=14 //Value=15
7 printf("%i\n",x); // Print=15 //Value=15
8 ++x; //Value=16
9 x++; //Value=17
10 printf("%i\n",x++); // Print=17 //Value=18
11 x++; //Value=19
12 printf("%i\n",++x); // Print=20 //Value=20
```

الفرق بين الـ Postfix & Prefix

لما يكون فى نفس الـ Statment أكثر من Instruction فى الحالة ده يظهر الفرق فى اولوية تنفيذ كل أمر

الفرق فى الـ Piority

6

7

Relational/Comparision Operators **False = 0 او True = 1** الناتج هنا دائما

==

>

<

>=

<=

!=



```
1 int x = 10;
2 int y = 20;
3 int z = 30;
4 int a = 10;
5
6 printf("x > y? %i\n",x > y);//0
7 printf("x < y? %i\n",x < y);//1
8 printf("x == y? %i\n",x == y);//0
9 printf("x == a? %i\n",x == a);//1
10 printf("x != a? %i\n",x != a);//0
11 printf("x >= a? %i\n",x >= a);//1
12 printf("x <= a? %i\n",x <= y);//1
```

8

Logical Operator **False = 0 او True = 1** الناتج هنا دائما

&&

كل الشروط لابد ان تتحقق، في حالة فشل اي شرط من الشروط تفشل العملية بأكملها
لازم كل الشروط تبح علشان العملية تبح

||

لا بد ان يتحقق شرط واحد فقط على الاقل لنجاح العملية، العملية كلها تفشل في حالة فشل كل الشروط

!

بتعكس اللوجيك



```
1 int age = 15;
2 char gender = 'm';
3 int salary = 5000;
4 int id = 123;
5 printf("%i\n", (age >=15 && gender == 'm' && salary>4000 && id == 123));//1
6 printf("%i\n", (age >=15 && gender == 'm' && salary>4000 && id == 1));//0
7
8 ///
9 printf("%i\n", (age >=18 || gender == 'f' || salary>7000 || id == 123));//1
10 printf("%i\n", (age >=18 || gender == 'f' || salary>7000 || id == 1));//0
11
12
13 printf("%i\n", !(age >=15 && gender == 'm' && salary>4000 && id == 123));//0
14 printf("%i\n", !(age >=15 && gender == 'm' && salary>4000 && id == 1));//1
15
16 ///
17 printf("%i\n", !(age >=18 || gender == 'f' || salary>7000 || id == 123));//0
18 printf("%i\n", !(age >=18 || gender == 'f' || salary>7000 || id == 1));//1
19
20 printf("%i\n", ((age >=15 && gender == 'm') || (salary>7000 && id == 123)));//1
```

9

Byte = 8Bits

Bitwise

&

|

~ Search

128	64	32	16	8	4	2	1		
2P7	2P6	2P5	2P4	2P3	2P2	2P1	2P0		
0	0	0	0	1	0	1	0	10	x
0	0	0	0	1	1	1	1	15	y
0	0	1	1	1	1	0	1	61	z
0	0	0	0	1	0	1	0	10	& x & y
0	0	0	0	1	1	1	1	15	x y
0	0	1	1	1	1	1	1	63	y z

10

Control Flow

• Conditional/Decision Statments

الجملة الشرطية

- if Condition
- Else
- Switch

```
if (Condetion's){  
    //Logic  
  
}else{  
    //logic  
}
```

if Constion True Will Execute Body

Logical Error
And Very Bad Performance

```
1 //v1  
2 int grade = 90;  
3 if(grade >= 90){  
4     printf("Excellent\n");  
5 }  
6 if(grade >=80){  
7     printf("Very Good\n");  
8 }  
9 if(grade >=70){  
10    printf("Good\n");  
11 }  
12 if(grade >=60){  
13    printf("Fair\n");  
14 }else {  
15    printf("Filed\n");  
16 }
```

```
v2  
int grade = 100;  
if(grade >= 90 && grade <=100)  
    printf("Excellent\n");  
else  
if(grade >=80 && grade <90)  
    printf("Very Good\n");  
else  
if(grade >=70 && grade <80)  
    printf("Good\n");  
else  
if(grade >=60 && grade <70)  
    printf("Fair\n");  
else  
if(grade<60)  
    printf("Filed\n");  
}
```

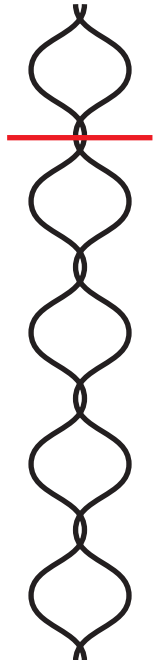
Very Bad Performance

```
1 //v2  
2 int grade = 90;  
3 if(grade >= 90 && grade <=100){  
4     printf("Excellent\n");  
5 }  
6 if(grade >=80 && grade <90){  
7     printf("Very Good\n");  
8 }  
9 if(grade >=70 && grade <80){  
10    printf("Good\n");  
11 }  
12 if(grade >=60 && grade <70){  
13    printf("Fair\n");  
14 }  
15 if(grade<60) {  
16    printf("Filed\n");  
17 }
```

```
int grade = 100;  
if(grade >= 90 && grade <=100){  
    printf("Excellent\n");  
}  
else if(grade >=80 && grade <90){  
    printf("Very Good\n");  
}  
else if(grade >=70 && grade <80){  
    printf("Good\n");  
}  
else if(grade >=60 && grade <70){  
    printf("Fair\n");  
}  
else {  
    printf("Filed\n");  
}
```

Very Good Performance

```
1 //v3  
2 int grade = 100;  
3 if(grade >= 90 && grade <=100){  
4     printf("Excellent\n");  
5 }  
6 else if(grade >=80 && grade <90){  
7     printf("Very Good\n");  
8 }  
9 else if(grade >=70 && grade <80){  
10    printf("Good\n");  
11 }  
12 else if(grade >=60 && grade <70){  
13    printf("Fair\n");  
14 }  
15 else {  
16    printf("Filed\n");  
17 }
```



اي قيمة غير 0 هي تمثل true

```
1 int x =10;
2 int y=20;
3
4 if(x){
5     printf("True\n");//True
6 }
7
8 if(x=y){
9     printf("True\n");//True
10 }
11
12 if(1){
13     printf("True\n");//true
14 }
15 if(!0){
16     printf("True\n");
17 }
18 if(!1){
19     printf("True\n");//true
20 }
21 if(0){
22     printf("True\n");
23 }
24
25 if(x++){
26     printf("True\n");//true
27 }
28
29 if(x+1){
30     printf("True\n");//true
31 }
```

Swtich

```
Switch(Variable){
    case Val:
        //logic

    break;
    case val2:
        //logic
        break;

    default:
        //logic
        break;
}
```

```
1 switch(Day){
2     case 1:
3         printf("FriDay\n");
4         break;
5     case 2:
6         printf("Sat\n");
7         break;
8     case 3:
9         printf("Sun\n");
10        break;
11    case 4:
12        printf("Mon\n");
13        break;
14    case 5:
15        printf("Tues\n");
16        break;
17    case 6:
18        printf("Wid\n");
19        break;
20    default:
21        printf("Thurs\n");
22        break;
23 }
```

شروط Switch

تستخدم فقط في حالة المقارنة من حيث Equality

يعمل على خاصية Indexing

يجب وضع قيمة صحيحة وليست محفوظة في متغيرات

تعمل فقط على نوعين من البيانات هما: Int & Char

```
int Day = 2
switch(Day){
    case 1:
        printf("FriDay\n");
        break;
    case 2:
        printf("Sat\n");
        break;
    case 3:
        printf("Sun\n");
        break;
    case 4:
        printf("Mon\n");
        break;
    case 5:
        printf("Tues\n");
        break;
    case 6:
        printf("Wid\n");
        break;
    default:
        printf("Thurs\n");
        break;
}
```

Fall Through

هينفذ جميع الاوامر من بعد الشرط الصحيح
حتى يصتدم بأول Break

Fall Out

```
1 int Day = 1;
2 switch(Day){
3     case 1:
4         printf("FriDay\n");
5
6     case 2:
7         printf("Sat\n");
8
9     case 3:
10        printf("Sun\n");
11
12    case 4:
13        printf("Mon\n");
14
15    case 5:
16        printf("Tues\n");
17        break;
18    case 6:
19        printf("Wid\n");
20        break;
21    default:
22        printf("Thurs\n");
23        break;
24 }
```

End Of Switch

End Of Switch

Muly Choice User Fall Through Concept

```
1 char Color = 'r';
2 switch(Color){
3
4     case 'r':
5     case 'R':
6     case 'q':
7         printf("Red\n");
8         break;
9     case 'g':
10        printf("Green\n");
11        break;
12    case 'b':
13        printf("Blue\n");
14        break;
15 }
```

```
1 int Day = 2
2 switch(Day){
3     case 1:
4         printf("FriDay\n");
5         break;
6     case 2:
7         printf("Sat\n");
8         break;
9     case 3:
10        printf("Sun\n");
11        break;
12    case 4:
13        printf("Mon\n");
14        break;
15    case 5:
16        printf("Tues\n");
17        break;
18    case 6:
19        printf("Wid\n");
20        break;
21    default:
22        printf("Thurs\n");
23        break;
24 }
```