CMSC 104 - Lecture 13 C Grasso and the Internet

Review Questions

#1 - Question

```
#include<stdio.h>
void main(){
```

```
int a=5, b=10, c=1;
```

```
if ( a&&b > c ) {
```

```
printf("cquestionbank");
}
else {
    break;
}
```

- (A) cquestionbank
- (B) It will print nothing
- (C) Run time error
- (D) Compilation error
- (E) None of the above

#1 – Explanation

Keyword break is not syntactical part of ifelse statement. So we cannot use break keyword in if-else statement. This keyword can be use in case of loop or switch case statement. Hence when you will compile above code compiler will show an error message: Misplaced break.

#2 - Question

```
#define PRINT printf("Star
Wars"); printf(" Psycho");
```

```
#include<stdio.h>
```

```
void main(){
```

}

```
int x=1;
if(x--)
PRINT
else
printf("The Shawshank
Redemption");
```

- (A) Stars Wars Psycho
- (B) The Shawshank Redemption
- (C) Warning: Condition is always true
- (D) Warning: Condition is always false
- (E) Compilation error

#2 – Explanation

PRINT is macro constant. Macro PRINT will be replaced by its defined statement just before the actual compilation starts. Above code is converted as:

```
void main(){
    int x=1;
    if(x--)
        printf("Star Wars");
        printf(" Psycho");
        else
        printf("The Shawshank Redemption");
}
```

#2 – Explanation

If you are not using opening and closing curly bracket in if clause, then you can write only one statement in the if clause. So compiler will think:

(A) It is if statement without any else. It is ok.

```
if(x--)
    printf("Star Wars");
```

(B) It is a function call. It is also ok

```
printf(" Psycho");
```

#2 – Explanation

(C) You cannot write else clause without any if clause. It is cause of compilation error. Hence compiler will show an error message: Misplaced else

else
 printf("The Shawshank Redemption");

#3 - Question

#define True 5==5

```
#include<stdio.h>
```

```
void main() {
```

```
if ( .001-0.1f )
    printf("David Beckham");
else if(True)
    printf("Ronaldinho");
else
```

```
printf("Cristiano Ronaldo");
```

Choose all that apply:

- (A) David Beckham
- (B) Ronaldinho
- (C) Cristiano Ronaldo
- (D) Warning: Condition is always true
- (E) Warning:

Unreachable code



As we know in C zero represents false and any non-zero number represents true. So in the above code:

- (A) (0.001 0.1f) is not zero so it represents true. So if clause will execute and it will print:
 David Beckham on console.
- (D) But it is bad programming practice to write constant as a condition in if clause. Hence compiler will show a warning message:
 - Condition is always true
- (E) Since condition is always true, so else clause will never execute. Program control cannot reach at else part. So compiler will show another warning message:
 - Unreachable code



```
#include<stdio.h>
void main(){
    int a=100;
    if ( a>10 )
        printf("M.S. Dhoni");
    else if ( a>20 )
        printf("M.E.K Hussey");
    else if(a>30)
        printf("A.B. de ville");
}
```

- (A) M.S. Dhoni
- (B) A.B. de villiers
- (C) M.S Dhoni M.E.K Hussey A.B. de Villiers
- (D) Compilation error: More than one conditions are true
- (E) None of the above

#4 – Explanation

- In case of if if else if else …
 Statement
- If the first if clause is true, the compiler will never check rest of the if-else clause and so on.

#5 - Question

#include<stdio.h>

```
void main(){
    int x = -1, y = -1;
    if ( ++x = ++y )
        printf("R.T. Ponting");
    else
        printf("C.H. Gayle");
}
```

Choose all that apply:
(A) R.T Ponting
(B) C.H. Gayle
(C) Warning: x and y are assigned a value that is never used
(D) Warning: Condition is always true
(E) Compilation error

#5 – Explanation

Consider following statement: ++x=++y

As we know ++ is pre increment operator in the above statement. This operator increments the value of any integral variable by one and return that value. After performing pre increments above statement will be:

- 0=0
- In C language it is illegal to assign a constant value to another constant. Left side of = operator must be a container i.e. a variable. So compiler will show an error message: Lvalue required
- In C if you assign any value to variable but you don't perform any operator or perform operation only using unary operator on the variable the complier will show a warning message: Variable is assigned a value that is never

#6 - Question

#include<stdio.h>

```
void main(){
    if ( sizeof(void) )
        printf("M. Muralilidar");
    else
        printf("Harbhajan Singh");
}
```

- (A) M. Muralilidaran
- (B) Harbhajan Singh
- (C) Warning: Condition is always false
- (D) Compilation error
- (E) None of the above

#6 – Explanation

(B) Harbhajan Singhsizeof(void) is 0



#include<stdio.h>

```
void main(){
    int m=5,n=10,q=20;
    if( q/n*m)
        printf("William Gates");
    else
        printf(" Warren Buffet");
        printf(" Carlos Helu");
}
```

- (A) William Gates
- (B) Warren Buffet Carlos Helu
- (C) Run time error
- (D) Compilation error
- (E) None of the above



Consider the following expression: **q** / **n** * **m**

In this expression there are two operators. They are: /: Division operator and *: Multiplication operator

Precedence of both operators is same. Hence associate will decide which operator will execute first. Since Associate is left to right. So / operator will execute then * operator will execute.

#7 – Explanation

- Since 10 is non-zero number if clause will execute and print:
 - William Gates
- else clause there is not any opening and closing curly bracket, so compiler will treat only one statement as a else part. Hence last statement i.e. printf(" Carlos Slim Helu"); is not part of if-else statement. So at the end compiler will also print:
 - Carlos Slim Helu

#8 - Question

#include<stdio.h>

```
void main(){
    if(!printf("Mukesh Ambani"))
    if(printf(" Lakashmi Mittal"));
}
```

- (A) Mukesh Ambani
- (B) Lakashmi Mittal
- (C) It will print nothing
- (D) Mukesh Ambani Lakashmi Mittal
- (E) Compilation error: if statement without body

#8 – Explanation

- Return type of printf function is int. This function returns an integral value which is equal to number of characters a printf function will print on console.
- First of all printf function will print: Mukesh Ambani.
 Since it is printing 13 character it will return 13. So,

```
!printf("Mukesh Ambani")
= !13
= 0
```

 In C language zero represents false. So if(0) is false so next statement which <u>inside</u> the body of first if statement will <u>not</u> execute.

#9 - Question

```
#include<stdio.h>
void main(){
    if("ABC") printf("Obama\n");
    if(-1) printf("Jintao\n");
    if(.92L) printf("Sarkozy\n");
    if(0) printf("Bernanke\n");
    if('W') printf("Putin\n");
}
```

- (A) It will print nothing
- (B) Obama Jintao
 - Sarkozy
 - Putin
- (C) Obama Jintao Sarkozy
 - Bernanke
 - Putin
- (D) Jintao Sarkozy Putin
- (E) Compilation error

#9 – Explanation

- "ABC": It is string constant and it will always return a non-zero memory address.
- 0.92L: It is long double constant.
- 'W': It is character constant and its ASCII value is > 0
- In this program condition of first, second, third and fifth if statements are true.

#10 - Question

#include<stdio.h>

```
void main(){
    if(0xA)
        if(052)
            if('\xeb')
            if('\012')
                printf("Tom");
                else;
            else;
```

Choose all that apply: (A) Tom (B) Compilation error: Misplaced else (C) Compilation error: If without any body (D) Compilation error: Undefined symbol (E) Warning: Condition is always true

#10 – Explanation

oxA: It is hexadecimal integer constant. o52: It octal integer constant. '\xeb': It is hexadecimal character constant. '\o12': It is octal character constant.

- As we know in c zero represents false and any non-zero number represents true. All of the above constants return a non-zero value. So all if conditions in the above program are true.
- In c it is possible to write else clause without any body.

#11 - Question

#include<stdio.h>

```
void main(){
    int a=10;
    if(printf("%d",a>=10)-10)
        for(;;)
            break;
    else;
}
```

Choose all that apply:
(A) It will print nothing
(B) 0
(C) 1
(D) Compilation error: Misplaced else
(E) Infinite loop

#11 – Explanation

- Return type of printf function is int. This function return a integral value which is equal to number of charcters printf function will print on console.
- Since -9 is non-zero number so if(-9) is true condition hence if clause will execute an infinite for loop but due to break keyword it will come out of loop.