Quiz

Total points 45/50

The respondent's email (**memoahmedrabee@gmail.com**) was recorded on submission of this form.

0 of 0 points

Full name *

Marawan Ahmed

Qestions

45 of 50 points

✗ what is the problem with the following * 0/2
int 3x=10;

- (•) var name can't contain numers ✗
- ( ) no problem
- ( ) var name can't start with letter
- ( ) var name can't start with number

Correct answer

- (•) var name can't start with number

✓ What is the code output? * 2/2

```
#include<stdio.h>
int main(void)
{
    for(int i=0;i<5;i++); printf("Hello");
}
```

- ( ) HelloHelloHelloHelloHello
- (•) Hello ✓
- ( ) Compiler error

What is the code output? *

2/2

```
#include<stdio.h>
int main(void)
{
    switch(printf("Go "))
    {
        case 0: printf("Home"); break;
        case 3: printf("ahead"); break;
        case 2: printf("Back"); break;
        default: printf("sleeping"); break;
    }
}
```

- Home
- Go sleeping
- Go ahead ✓
- Go Home
- Go Back

Which loop is most suitable to first perform the operation and then test the condition? *

2/2

- for
- while
- do while ✓
- non of the above

#include<stdio.h> *

2/2

```
int main(void)
{
    int x=1;
    if(x>0)
        x--;
        printf("%d",x);
    else
        printf("%d",x);
}
```

- 1
- 0
- Warning
- Compilation error ✓

✓ What is the code output? * 2/2

```
#include<stdio.h>
int main(void)
{
    int x=5;
    int y=x++;
    printf("%d %d",y,x);
    y=--x;
    printf("/%d %d",y,x);
}
```

- (●) 56/55 ✓
- ( ) 66/65
- ( ) 56/65
- ( ) 65/45

✓ which of the following operations isn't allowed on pointers * 2/2

- ( ) Adding constant value to pointer
- ( ) substracting two pointers
- (●) Adding two pointers ✓
- ( ) substarcting constant value from pointer

✓ What is the code output? * 2/2

```
#include<stdio.h>
int main()
{
    int x=0;
    do{
        x++;
        printf("%d ,",x);
    }while(x<3);
}
```

- ( ) 1,2,3
- ( ) 1,2,
- (●) 1,2,3, ✓
- ( ) 1,2

What is the code output? *

2/2

```
#include<stdio.h>
void func1(int i);
void func2(int j);
int main()
{
    int i=3;
    func1(i);
    printf("%d",i);
    func2(i);
    printf("%d",i);
}
void func1(int i)
{
    printf("%d",++i);
}
void func2(int j)
{
    printf("%d",++j);
}
```

- 4,3,4,4
- 4,3,4,3 ✓
- 4,4,4,4
- 3,4,3,4

2/2

```
#include<stdio.h>
int main()
{
    int i=1;
    switch(i)
    {
        case 1: printf("1");
        case 2: printf("2"); break;
    }
}
```

- 1
- 2
- 12 ✓
- Compilation error

✓ #include<stdio.h> * 2/2

```
int main(void)
{
    signed char x=230;
    printf("%d",x);
}
```

- ○ 230
- ○ -25
- ◉ -26 ✓
- ○ -27

✓ #include<stdio.h> * 1/1

```
int main(void)
{       int i=0 , j=1 , k=2 , m;
        m= i++ && j++ || ++k;
        printf("m=%d , i=%d , j=%d , k=%d",m,i,j,k);
}
```

- ○ 1,1,1,2
- ○ 1,0,1,2
- ○ 1,1,2,3
- ◉ 1,1,1,3 ✓

✓ The symbol table is an output of the ..... tool * 2/2

- ○ Preprocessor
- ◉ Compiler ✓
- ○ Assembler
- ○ Linker

✓ ? Which of the following is a constant pointer to integer * 2/2

- ○ Const int const *ptr
- ○ const int *ptr
- ◉ int *const ptr ✓
- ○ int ptr

✓ Which bitwise operator is suitable for checking whether a particular bit is 1 or 0? *2/2

- (●) & operator ✓
- ( ) && operator
- ( ) || operator
- ( ) ! operator

✓ Which array definition has a correct syntax? * 2/2

- ( ) int array[5] = {0} ;
- ( ) int array[ ] = {1,3,8 };
- ( ) int array[5] = {[3]=4,[2]=2};
- (●) All of the above ✓

✓ The pointer itself isn’t a variable and it doesn't have an address in the memory; it only holds an address of another variable *2/2

- ( ) True
- (●) False ✓

✓ . What is the problem in the following C declarations? * 2/2
int func(int);
double func(int);
int func(float);

- ( ) A function with same name cannot have different return types
- ( ) A function with same name cannot have different number of parameters
- (●) We can not having more than function with the same name in C ✓

✓ ?Which of the Following is a volatile memory * 2/2

- (●) SRAM ✓
- ( ) FLASH
- ( ) EPROM
- ( ) ROM

✓ what the output of: * 2/2

```
int main(void) {
uint32_t x = 2;
uint8_t* ptr1 = &x;
static uint32_t i = x;
printf("%i, %i, %d\n",*ptr1, i, *x);
}
```

- ○ 2,2,2
- ◉ Compilation error ✓

✗ ... ? What is the output of the program given below * 0/1

```
#include<stdio.h>
struct point
{
int x ;
int y ;
} ;
void fun (struct point p[ ])
{
    printf("%d", p[1].x) ;
}
void main(void)
{
    struct point p1 [ ]= { {1 , 2} , {3 , 4 } } ;
    fun(p1) ;
}
```

- ◉ 1 ✗
- ○ 2
- ○ 3
- ○ Compilation error

Correct answer

- ◉ 3

✓ The range of signed char is from ..... to ..... but the range of unsigned char from ..... to ..... (given that the character size is assumed to be 1 byte) *2/2

- ○ -128 to 127 & 0 to 256
- ○ 0 to 255 & -128 to 127
- ○ -255 to 255 & 0 to 255
- ◉ -128 to 127 & 0 to 255 ✓

✓ Given the following line of code, what is correct about it ... ? Const int *ptr; *2/2

- (●) We cannot change the value pointed by ptr ✓
- ( ) We cannot change the pointer ptr itself
- ( ) Both of the above
- ( ) We can change the pointer as well as the We can change the pointer as well as the value pointed by it.

✗ Uninitialized global variables stored in ..... RAM * 0/2

- ( ) .bss
- ( ) .stack
- ( ) .data
- (●) .text ✗

Correct answer

- (●) .bss

✓ what is the size of this struct? * 2/2

```
struct   mytruct
}
char *x;
int y;
;{
```

Note: ur address bus is 64-bit.

- ( ) 12
- ( ) 5
- ( ) 8
- (●) 16 ✓

✓ #include int main() { *

int a = 3, b = 5;

int t = a;

a = b;

b = t;

printf("%d %d", a, b);

return 0;}

2/2

- 3 5
- 5 3 ✓
- 5 5
- 3 3