Question 1 of 10

What will be the output of the C program?

```
#include<stdio.h>
#define i 10
int main()
{
        #define i 20
        printf("%d",i);
        return 0;
}
```

- Compilation error
- 20
- 30
- Runtime error

Question 2 of 10

What will be output of following code?

```
#include<stdio.h>
#define p 11+2
int main()
{
    int i;
    i=p++ * p++;
    printf("%d",i);
    return 0;
}
```

- 35
- 26
- 169
- lvalue required error

Question 3 of 10

```
#include<stdio.h>
#define MAX(x, y) if(x-=x)x-=x;else y-=y
int main()
{
    int a=2,b=4;
    MAX(a,b);
    printf("%d %d\n", a,b);
    return 0;
}
```

- 0 0
- 2 4
- 4 4
- 3 2

Question 4 of 10

What is the output of C program

```
#include <stdio.h>
#define p 24;
int main()
{
  printf("%d",p);
  return 0;
}
```

- Garbage value
- Runtime error
- 24
- Compilation error

What is the return type of malloc() or calloc()?

- (●) int *
- ( ) int **
- ( ) void *
- ( ) void **

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
char *ptr=NULL;
int alloc_length =16 ;
int new_length;

ptr = (char *)malloc(sizeof(char)*alloc_length);

ptr = realloc(ptr, alloc_length << 1);

free(ptr);
ptr=NULL;

return 0;
}
```

- ( ) allocated 16bytes and free 32bytes
- ( ) allocated 32bytes and free 32bytes
- ( ) allocated 32bytes and free 16bytes
- ( ) allocated 16bytes and free 16bytes

Point out the correct statement which correctly free the memory pointed to by 'name' and 'e' in the following program and there will not be any memory leakage?

0 : 04 : 00

```
#include<stdio.h>
#include<stdlib.h>
struct emp
    {
        int ID;
        float dept;
        char *name;
    };

int main()
{
    struct emp *e;
    e = (struct emp *)malloc(sizeof(struct emp));
    e->name = (char*)malloc(20);
    return 0;
}
```

- free(e); , free(e->s);
- free(e->s); , free(e);
- free(e->s);
- free(e);

Question 9 of 10

what will be the output of this code?

```
#include<stdio.h>
#define a 100
int main()
{
printf("a : %d\n",a);
#ifndef a
#define a 30
#else
#undef a
#define a 40
printf("a : %d",a);
return 0;
}
```

- a : 100 a : 30
- a : 0 a : 100
- a : 100 a : 40
- Compile time error

what will be the output of this code?

```
#include<stdio.h>
#define a 10
#define z(a) a-1
#define SQR(x) (x*x)
int main()
{
int c, b=3;
c = z(a) * SQR(b+2);
printf("%d\n", c);
return 0;
}
```

- -1
- 225
- 99
- 5

Question 10 of 10

what will be the output of this code?

```
#include <stdio.h>
#define EQUAL(X, Y) X == Y
int main()
{
    #if EQUAL(X, 0)
        printf("SUNBEAM");
    #else
        printf("TEST");
    #endif
    return 0;
}
```

- ○ TEST
- ○ SUNBEAM
- ○ compile time error.
- ○ Error: in macro substitution

1. What will be the output of the C program?

```
#include<stdio.h>
#define i 10
int main()
{
        #define i 20
        printf("%d",i);
        return 0;
}
```

Answers

1. Compilation error

2. 20

3. 30

4. Runtime error

2. What will be output of following code?

```
#include<stdio.h>
#define p 11+2
int main()
{
    int i;
    i=p++ * p++;
    printf("%d",i);
    return 0;
}
```

Answers

1. 35

2. 26

3. 169

4. lvalue required error

3.

```
#include<stdio.h>
#define MAX(x, y) if(x-=x)x-=x;else y-=y;
int main()
{
    int a=2,b=4;
    MAX(a,b);
    printf("%d %d\n", a,b);
    return 0;
}
```

Answers

1. 0 0

2. 2 4

3. 4 4

4. 3 2

4. What is the output of C program

```
#include <stdio.h>
#define p 24;
int main()
{
  printf("%d",p);
  return 0;
}
```

Answers

1. Garbage value

2. Runtime error

3. 24

4. Compilation error

5. What is the return type of malloc() or calloc()?

Answers

1. int *

2. int **

3. void *

4. void **

6.
```
#include<stdio.h>
#include<stdlib.h>
int main()
{
char *ptr=NULL;
int alloc_length =16 ;
int new_length;

ptr = (char *)malloc(sizeof(char)*alloc_length);

ptr = realloc(ptr, alloc_length << 1);

free(ptr);
ptr=NULL;

return 0;
}
```

Answers

1. allocated 16bytes and free 32bytes

2. allocated 32bytes and free 32bytes

3. allocated 32bytes and free 16bytes

4. allocated 16bytes and free 16bytes

7. Point out the correct statement which correctly free the memory pointed to by 'name' and 'e' in the following program and there will not be any memory leakage?

```
#include<stdio.h>
#include<stdlib.h>
struct emp
    {
        int ID;
        float dept;
        char *name;
    };

int main()
{
    struct emp *e;
    e = (struct emp *)malloc(sizeof(struct emp));
    e->name = (char*)malloc(20);
    return 0;
}
```

Answers

1. free(e); , free(e->s);

2. free(e->s); , free(e);

3. free(e->s);

4. free(e);

8. what will be the output of this code?

```
#include<stdio.h>
#define a 100
int main()
{
printf("a : %d\n",a);
#ifndef a
#define a 30
#else
#undef a
#define a 40
printf("a : %d",a);
return 0;
}
```

Answers

1. a : 100 a : 30

2. a : 0   a : 100

3. a : 100 a : 40

4. Compile time error

9. what will be the output of this code?

```
#include<stdio.h>
#define a 10
#define z(a) a-1
#define SQR(x) (x*x)
int main()
{
int c, b=3;
c = z(a) * SQR(b+2);
printf("%d\n", c);
return 0;
}
```

Answers

1. -1

2. 225

3. 99

4. 5

10. what will be the output of this code?

```
#include <stdio.h>
#define EQUAL(X, Y) X == Y
int main()
{
    #if EQUAL(X, 0)
        printf("SUNBEAM");
    #else
        printf("TEST");
    #endif
    return 0;
}
```

Answers

1. TEST

2. SUNBEAM

3. compile time error.

4. Error: in macro substitution