1.

```c
#include<stdio.h>
#include<stdlib.h>
int main( void )
{
	int *sum = (int *)malloc(sizeof(int));
	sum = NULL;
	free(sum);
	return 0;
}
```

A. Compliation Error
B. Error free
C. Memory Leakage
D. dangling pointer
E. No output

Answer : C

2.

```c
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
int main(void)
{
	char *ptr=NULL;
	ptr = (char *)calloc(1,10);
	strcpy(ptr, "Sunbeam");
	ptr = (char *)realloc(ptr,20);
	strcat(ptr," IT PARK");
	printf("%c",(ptr[0]>=65 && ptr[0]<=90)  ?
			ptr[strlen(ptr)-1]+32 : ptr[strlen(ptr)-14]-32);
	free(ptr);
	ptr=NULL;

	return 0;
}
```

A. K
B. k
C. sfc
D. S
E. Error

Answer: B

3.

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
int main ( void )
{
	char *title=NULL;

	title = (char *) malloc(20);
	strcpy(title, "C Programming");
	printf("String = %c", *title);
	free(title); title=0;

	strcpy(title, "C++ Programming");
	printf(" %s", title);

	return 0;
}
```

A. String = C Programming
B. Complile time error
C. String = C C++ Programming
D. Run time error

Answer: D

4.

```
#include <stdio.h>
#define value 10
void fun();
int main( void )
{
	fun();
	return 0;
}
void fun()
{
	#ifndef value
		#undef value
		#define value 100
	#else
		#undef value
		#define value 200
	#endif

	#define Value 300
	printf("Value : %d",value);

	return ;
}
```

A. Value : 100
B. Value : 200
C. no output
D. Compile time error
E. Value : 300

Answer: B

5.

```
#include<stdio.h>
#define SWAP(a, b) {b ^= b; a ^= a; b ^= b ;}
int main( void )
{
	int x = 10;
	int y = 20;

	x=x*y; y=x/y; x=x/y;

	SWAP(x, y);

	x=x+y; y=x-y; x=x-y;

	printf("X=%d,Y=%d",x,y);

	return 0;
}
```

A. X=0,Y=0
B. X=10,Y=20
C. X=20,Y=10
D. Compile time error
E. X=10,Y=10

Answer: A