RECURSION

Exercises:

1)
```
int lenstr(char *str)
{
   if(*str=='\0')
        return 0;
   else
        return 1 + lenstr(str+1);
}

void main()
{
   char s[30]; int len;
   printf("Enter a string : ");
   gets(s);
   len=lenstr(s);
   printf("Length = %d", len);
}
```

2)
```
int HCF(int a, int b)
{
   if(b!=0)
        return HCF(b, a%b);
   else
```

```
        return a;
}

void main()
{
   int m, n;
   printf("Enter two numbers : ");
   scanf("%d %d", &m, &n);
   printf("HCF = %d", HCF(m, n));
}
```

3)
```
void Fib(int n)
{
   static int a=0, b=1, c;
   if(n>0)
   {
      c=a+b;
      a=b;
      b=c;
      printf("%d ",c);
      Fib(n-1);
   }
}

void main()
{
   int n;
   printf("Enter the limit : ");
   scanf("%d",&n);
   printf("%d %d ", 0, 1);
```

```
    Fib(n-2);
}
```

4)
```
int space(char c)
{
    if(c==32)
        return 1;
    else
        return 0;
}

void main()
{
    char s[50]; int i, c=0;
    printf("Enter a string : ");
    gets(s);
    for(i=0; s[i]!='\0'; i++)
        c+=space(s[i]);
    printf("No. of blanks : %d", c);
}
```

5)
```
void oct(int n)
{
        if(n>0)
        {
                oct(n/8);
                printf("%d", n%8);
        }
}
```

```
void main()
{
   int num;
   printf("Enter decimal equivalent of a number : ");
   scanf("%d", &num);
   printf(“Octal is : ”);
   oct(num);
}
```

6)
```
void hex(int n)
{
        if(n>0)
        {
                hex(n/16);
                if(n%16>=0 && n%16<=9)
                        printf("%d", n%16);
                else if(n%16>=10 && n%16<=15)
                {
                        switch(n%16)
                        {
                                case 10:
                                        printf("A");
                                        break;
                                case 11:
                                        printf("B");
                                        break;
                                case 12:
                                        printf("C");
                                        break;
                                case 13:
```

```
                        printf("D");
                        break;
                    case 14:
                        printf("E");
                        break;
                    case 15:
                        printf("F");
                }
            }
        }
    }
    void main()
    {
       int num;
       printf("Enter decimal equivalent of a number : ");
       scanf("%d", &num);
       printf("Hexadecimal is : ");
       hex(num);
    }
```

7)
```
    int fact(int n)
    {
       if(n>1)
           return n*fact(n-1);
       else
           return 1;
    }

    void main()
    {
```

```
    int num;
    printf("Enter a number : ");
    scanf("%d", &num);
    printf("Factorial = %d", fact(num));
}
```

8)
```
int sum(int n)
{
    if(n==0)
        return 0;
    return n%10+sum(n/10);
}

void main()
{
    int num;
    printf("Enter a number : ");
    scanf("%d", &num);
    printf("Sum of digits = %d", sum(num));
}
```

Aptitude Questions:

1) False
2) False
3) True
4) False
5) (IV) Until the stack overflows