```c
#include<stdio.h>

int s(int x)
{
 return x*x;

}

int c(int p)
{

return p*p*p;

}

void main()
{
 int x,ans;
 int (*p)(int);

char ch;
 x=5;
       printf("enter choice s,c");
       scanf("%c",&ch);

//pointing and calling function as per condition

 if(ch=='s')
 {     p=s;
         ans=p(x);
       printf("%d",ans);
  }

  else if(ch=='c')
       {
         p=c;
        ans=p(x);
        printf("%d",ans);
      }
}


#include<stdio.h>

// Note our user-defined comparison is the third parameter
void SelectionSort(int arr[], int nSize, int (*pComparison)(int,int))
{  int t;
      for (int i= 0; i <nSize-1; i++)
       {
          // Search through every element starting at nStartIndex+1
       for (int j = i+ 1; j <nSize; j++)
             {
                // Note that we are using the user-defined comparison here
           if (pComparison(arr[i], arr[j])) // COMPARISON DONE HERE
              {
                     t=arr[i];
                     arr[i]=arr[j];
                     arr[j]=t;
```

```
        }
           }
    }
}

// Here is a comparison function that sorts in ascending order
// (Note: it's exactly the same as the previous Ascending() function)
int Ascending(int nX, int nY)
{
return nX>nY;
}

// Here is a comparison function that sorts in descending order
int Descending(int nX, int nY)
{
return nX<nY;
}

// This function prints out the values in the array
void PrintArray(int pArray[], int nSize)
{
for (int iii=0; iii <nSize; iii++)
        printf("\n%d", pArray[iii] );

}

int main()
{
        int arr[9] = { 3, 7, 9, 5, 6, 1, 8, 2, 4 };

    // Sort the array in descending order using the Descending() function
    //SelectionSort(arr, 9, Descending);
    //PrintArray(arr, 9);

    // Sort the array in ascending order using the Ascending() function
SelectionSort(arr, 9, Ascending);
PrintArray(arr, 9);

return 0;
}
```

```
#include<stdio.h>

void Sqr(int x)

{       printf("%d",X*X);

}

void cube(int x)

{       printf("\n%d",X*X);

}
```

```
void increase(int x)
{                       printf("\n%d",++x);
}
int main()
{ void (*ptr[3])(); // ptr is an array of pointer to a function accepts integer and returns nothing
            ptr[0]=Sqr;
            ptr[1]=cube;
            ptr[2]=increase;
      for(int i=0;i<3;i++)
            {   ptr[i]();
            }
            return 0;
}
void main( )
     { int i;
   void fun1( ), fun2( ), fun3( );
        int (*f[3])( );
        f[0]=fun1;   f[1]=fun2;          f[2]=fun3;
          for(i=0;i<3;i++)
         (*f[i]) ( );
     }
     fun1( )   { printf(“USM”);        }
     fun2( )   { printf(“COMPUTER”);          }
     fun3( )   { printf(“EDUCATION”);         }
```

what will be the output

1) USM COMPUTER EDUCATION

2) USM COMPUTER

3) USM COMPUTERCOMPUTEREDUCATION

4) Error