1. (2 pt.)

```
all: program

program: file1.o file2.o file3.o file4.o
     g++ file1.o file2.o file3.o file4.o –o program

file1.o: file1.cpp
     g++ -c file1.cpp

file2.o: file2.cpp
     g++ -c file2.cpp

file3.o: file3.cpp
     g++ -c file3.cpp

file4.o: file4.cpp
     g++ -c file4.cpp

clean:
     rm -rf *o program
```

2. (3 pt.)
   a) 5
   b) Address of 6
   c) DE
   d) B
   e) EFG
   f) I

3. (2 pt.)

```
bool strEql (const char A[],  const char B[])
{
     int i=0;
     bool rval=false;
     while ((A[i]==B[i]) && (A[i]!='\0') && (B[i]!= '\0'))
          i++;
     if (A[i]== B[i])
          rval=true;

     return rval;
}
```

4. (3 pt.)

```
Node *insertNode (Node *list)
{
      Node *tmp = list; // temperay pointer to traverse linked list
      Node *newNode = new Node;
      cout <<" a integer value: "<<endl;
      cin >> newNode->info;
      // list is empty
      if (list ==NULL)
            list= newNode;

      else
      {
            //Find the last node or duplication
            while ((tmp->next != NULL) && ((tmp->info != newNode->info)
                  tmp = tmp->next;
            if (tmp->info== newNode->info)
            {
                  cout << newNode->info <<" is already in the list! "<<endl;
                  delete newNode;
            }
            else
                  //insert as the last element
                  tmp->next = newNode;

      }
      return list;
}
```