

Practice

Node sList;

NodePtr psList;

NodePtr *hsList;

```
typedef struct Node *NodePtr;
typedef struct Node
{
    char data;
    NodePtr psNext;
} Node;
```

// create an empty List

void lstCreate(NodePtr *hsList);

void lstInsertFirst(NodePtr *hsList, char data);

void lstDeleteFirst(NodePtr *hsList, char* pData);

void lstTerminate(NodePtr *hsList);

```
NodePtr psList;
int i;

lstCreate(&psList);

for(i = 0 ; i < 10 ; i++)
{
    lstInsertFirst(&psList, 'a' + i);
}
```


This code is seeded with errors!

```
void foo(int **hArg, int *pInt)
{
    *hArg = 12;

    *hArg = (int*) malloc(sizeof(int));

    **hArg = 13;

    *hArg = pInt;

    **hArg = 11;

    hArg = (int*) malloc(sizeof(int));

    return;
}

int main()
{
    int other = 7;
    int num = 9;
    int *pNum = & num;
    int *pVal = &other;

    foo(&pVal, pNum);

    printf("%d %d", *pVal, *pNum);

    free(                ); // ?????
    return EXIT_SUCCESS;
}
```