1. Are there any memory errors in the following programs? If so, list all of them. Assume that the user enters in correct input, and that the sizes entered are at least one.   
  
Write your solution in a text or Word file and submit it.

void main() {

char \*str, \*input;

int \*ilist;

int i, size1, size2;

printf("Number of letters in word: ");

scanf("%d", &size1); /\* user inputs an integer \*/

printf("Number of integers: ");

scanf("%d", &size2); /\* user inputs an integer \*/

str = (char \*) malloc(size1 \* sizeof(int));//有错，内存大小分配错误

ilist = (int \*) malloc(size2 \* sizeof(int));//有错，内存大小分配错误

printf("Word: ");

scanf("%s", str); /\* user inputs a string \*/

for(i = 0; i < size2; i++) {

printf("Number %d of %d: ", i + 1, size2);

scanf("%d", ilist + i); /\* user inputs an integer \*/

free(str);

str = NULL;

free(ilist);

ilist = NULL;

}

}

修改如上，malloc分配空间时没有乘以类型所占的空间大小，导致分配出来的空间不足,此外也没有进行内存释放，造成了内存泄漏。

2. Are there any memory errors in the following program? If so, list all of them.   
  
Write your solution in a text or Word file and submit it below.   
  
/\* return 1 if str is "1", 0 otherwise \*/  
int checkIf1(const char \*str) {

const char \*newstr = (char \*)malloc(strlen(str) + 1);

strcpy(newstr, str); /\* set newstr to str \*/

if (strcmp(newstr, "1") == 0) { /\* newstr is "1" \*/

free(newstr);

newstr = NULL;

return 1;

}

free(newstr);//内存泄漏

newstr = NULL;

return 0;

}  
void main() {

const char \*strArr[4] = {"1", "2", "3", "4"};

int i;

for(i = 0; i < 4; i++) {

printf("%d\n", checkIf1(strArr[i]));

}

}

修改如上，if里面未进行内存释放，导致了内存的泄漏，其他为了更安全，释放内存之后赋值为null，并在char前加了const。  
  
3. Are there any memory errors in the following program? If so, list all of them.   
  
Write your solution in a text or Word file and submit it below.   
  
struct data {

char \*str1, \*str2;

};   
  
/\* returns two strings concatenated if they are not the same, NULL otherwise \*/  
char \*mergeSingleIfDifferent(const char \*s1, const char \*s2) {

char \*str = (char \*) malloc(strlen(s1) + strlen(s2) + 1);  
if (strcmp(s1, s2) == 0) { /\* strings are equal \*/

free(str);

str = NULL;

}  
else {

strcpy(str, s1);

strcat(str, s2);

}  
return str;

}  
  
/\* copies merged strings (or NULL) into array of strings passed in (results) \*/  
void mergeArrayIfDifferent(char \*\*results[], const char \*strA1[], const char \*strA2[], int size) {

int i;  
  
for(i = 0; i < size; i++) {

results[i] = mergeSingleIfDifferent(strA1[i], strA2[i]);

}

}

void printAndFree(int c, char \*str) {

if (str != NULL) {

printf("%d: %s\n", c, str);  
free(str);

str = NULL;

}

}  
  
void main() {

const char \*strArr1[8] = {"1", "2", "3", "4", "5", "6", "7", "8"};  
const char \*strArr2[8] = {"a", "2", "c", "4", "e", "6", "g", "8"};  
const char \*results[8];  
int i;  
  
mergeArrayIfDifferent(&results, strArr1, strArr2, 8);  
for(i = 0; i < 8; i++) {

printAndFree(i, results);

}

}

修改如上，没有释放内存导致内存泄漏，同样为了更安全，指针赋值的都添加const。