Hacettepe University

BBM 203
Software Laboratory I

ASSIGNMENT 1

Fatih OKSUZOGLU
21328284

1-Problem Definition

In this problem definition, I solved using read-write files, pointers, matrices with multi dimensional arrays and dynamic memory allocation. I solved the problem using some of the conveniences in C language.

I divide the given scripts and direct them to the functions according to the command at the beginning of the line.If there is a create operation, I created a 2 dimensional array using dynemic memory allocation.I did the necessary operations on the matrices.Also, if the area of the matrix needs to be enlarged, I again reserved the area using dynemic memory allocation.

2-Methods and Solution

```
char *output = NULL;
int CharInFile = 0;
int c = 0;
do {
    c = 0;
    c = fgetc (ds);
    if (c != EOF) {
        output = (char *) realloc (output, (CharInFile + 2) * sizeof(char));
        output[CharInFile] = c;
    }
    CharInFile++;
} while (c != EOF);
```

The command file is being read as char using realloc. Then split as a line.

```
matrix * m = (matrix *) malloc(j*sizeof(matrix));
vector * v = (vector *) malloc(j*sizeof(vector));
for (int i = 0; i < j; ++i){
    command(m,v,CharInFile,j,aline[i],argv[1],fp);
}
```

I used matrix and vector arrays to store the data.

As shown below, the command function performs strcmp code control on lines separated by space.

```
int command(matrix*m,vector*v,int CharInFile,int j,char * command,char *in,FILE *fp){
    int k=0;
    char *bline[CharInFile];
    char *spaceline = strtok(command," ");
    while (spaceline!=NULL){
        bline[k]=spaceline;
        spaceline = strtok(NULL," ");
        k++;
    }
    if (strcmp(bline[0],"veczeros") == 0){ ...
    } else if (strcmp(bline[0],"matzeros")==0){ ...
    } else if (strcmp(bline[0],"vecread")==0){ ...
    } else if (strcmp(bline[0],"matread")==0){ ...
    } else if (strcmp(bline[0],"vecstack")==0){ ...
    } else if (strcmp(bline[0],"matstack")==0){ ...
    } else if (strcmp(bline[0],"mvstack")==0){ ...
    } else if (strcmp(bline[0],"pad")==0){ ...
    } else if (strcmp(bline[0],"padval")==0){ ...
    } else if (strcmp(bline[0],"vecslice")==0){ ...
    } else if (strcmp(bline[0],"matslicecol")==0){ ...
    } else if (strcmp(bline[0],"matslicerow")==0){ ...
    } else if (strcmp(bline[0],"matslice")==0){ ...
    } else if (strcmp(bline[0],"add")==0){ ...
    } else if (strcmp(bline[0],"multiply")==0){ ...
    } else if (strcmp(bline[0],"subtract")==0){ ...
    } else if (bline[0]!=NULL){ ...
    }
}
```

```
m[i].data = (int **) malloc(rows*sizeof(int*));
for(int x = 0; x < rows; x++){
    m[i].data[x] = (int*)calloc(cols, sizeof(int));
}
```

```
v[i].data = (int *) malloc(len1*sizeof(int));
```

I used malloc and calloc to create a new matrix when executing the read matrix and zero matrix commands.In the vector commands, I only used malloc.

```
m[matlindex].data = (int **) realloc (m[matlindex].data,m[matlindex].rows * sizeof(int*));
for(int x = 0; x < m[matlindex].rows; x++){
    m[matlindex].data[x] = (int*)calloc(m[matlindex].cols, sizeof(int));
}
```

I used realloc for commands like pad, padval, matstack, mvstack. I was able to expand my matrix and make extra space available.

```
/* Finding the index of the vector in the vector array. */
int indevec(vector * v,char *name) {    ...
}
/* Finding the index of the Matrix in the matrix array */
int indemat(matrix * m,char *name){ ...
}
```

In addition, I created indevec and indemat functions. As written, these functions return the index of the given vector or matrix.

3-Functions implemented

Initialization Functions

I used malloc directly in vectors for initialization. In the matrices, I preferred to use calloc for the columns while allocating space with malloc for each row.

In reading functions, instead of writing directly into it, I first created a vector and matrix consisting of zeros with zero functions. Then I placed the data in them.

While doing all of this, I suppressed the error command to prevent the file from being present or to have a data of this name before.

Concatenation Functions

In concatenation functions, I first combined the data into an array. Then, with the Zero Matrix function, I created and wrote the matrix that the function asked us to create.

If this is an overwrite, I chose to release the matrix data first and then fill it again.

While doing all of this, I suppressed the error command if no matrix is found or if the parameters are entered incorrectly.

Padding Functions

When writing pad and padval functions, I followed the path in concatenation functions. I threw the data in the matrix into an array. Then I applied pad and padval operations. I have written the array formed with the new data on the new matrix using the zeromatris function.

Particularly in the pad function, I took care to control both horizontally and vertically.

Slicing Functions

When the dimensions of the slicing functions are given as parameters, I created directly with zeromatrix.

If the parameters are moving outside of the dimensions, I have given an error.

Math Functions

While applying mathematical functions, I applied matrix1 operations and wrote on it as in other functions.I've paid attention to the compatibility of the rows and columns. I wrote an error when it was incompatible.

Extra Functions

bigone,smallone : Comparative, pad auxiliary function.
printVector, printMatrix : Prints the function in the given index.
indevec , indemat : In all functions, I check the existence of the given matrix and vector functions.
freematrix, freevector : This function is used to free all data.

REFERENCES

[1] "Dynamic Memory Allocation in C".
https://www.geeksforgeeks.org/dynamic-memory-allocation-in-c-using-malloc-calloc-free-and-realloc/ [Accessed: Nov 09, 2019].
[2] "Dinamik Bellek Yönetimi".
http://www.cemdemir.net/c-programlama-dili/dinamik-bellek-yonetimi-321.html
[Accessed: Nov 08, 2019].
[3] "Dynamically allocate a 2D array in C".
https://www.geeksforgeeks.org/dynamically-allocate-2d-array-c/ [Accessed: Nov 09, 2019].
[4] "Realloc function examples in C".
https://stackoverflow.com/questions/13748338/how-to-use-realloc-in-a-function-in-c/ [Accessed: Nov 09, 2019].