

Spotting up Labels & comments

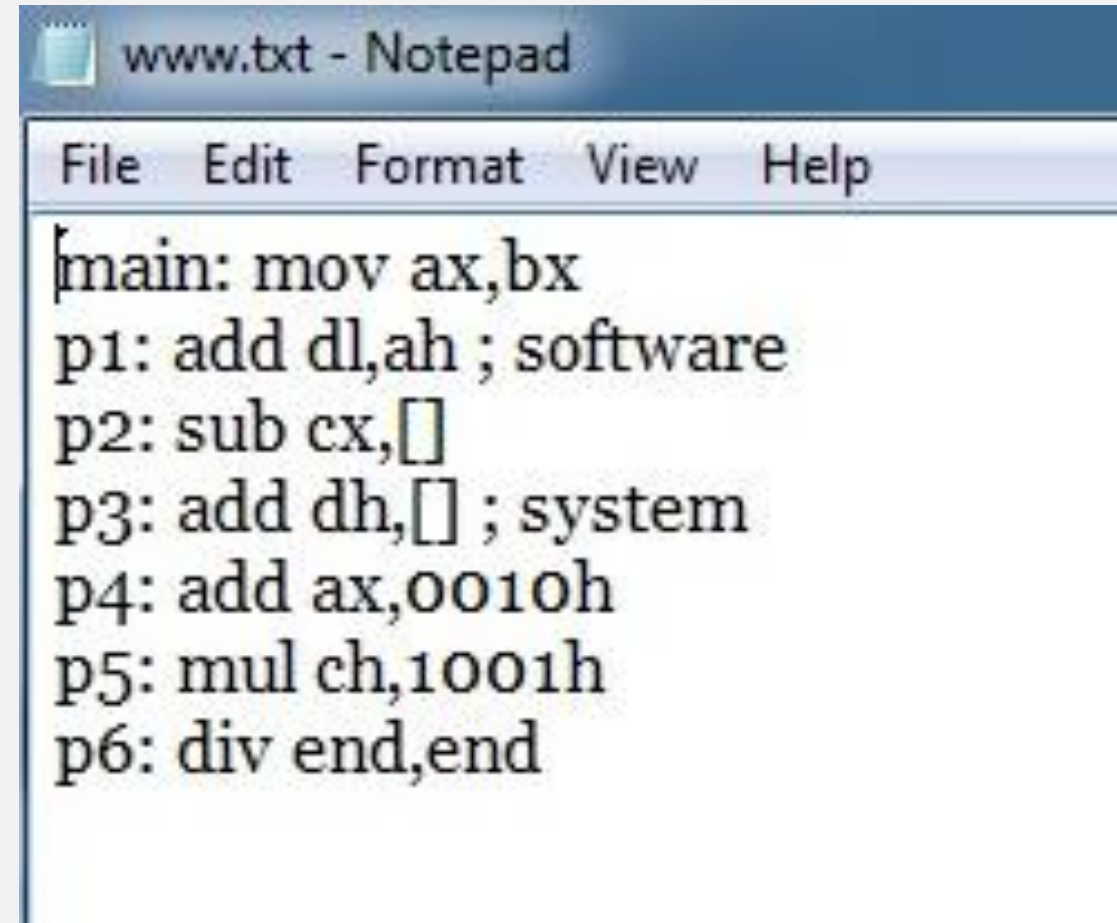
PASS I ASSEMBLER-PART I

Course2- Week-3

PASS1 ASSEMBLER

Following are the Requirements for programming the pass I assembler:

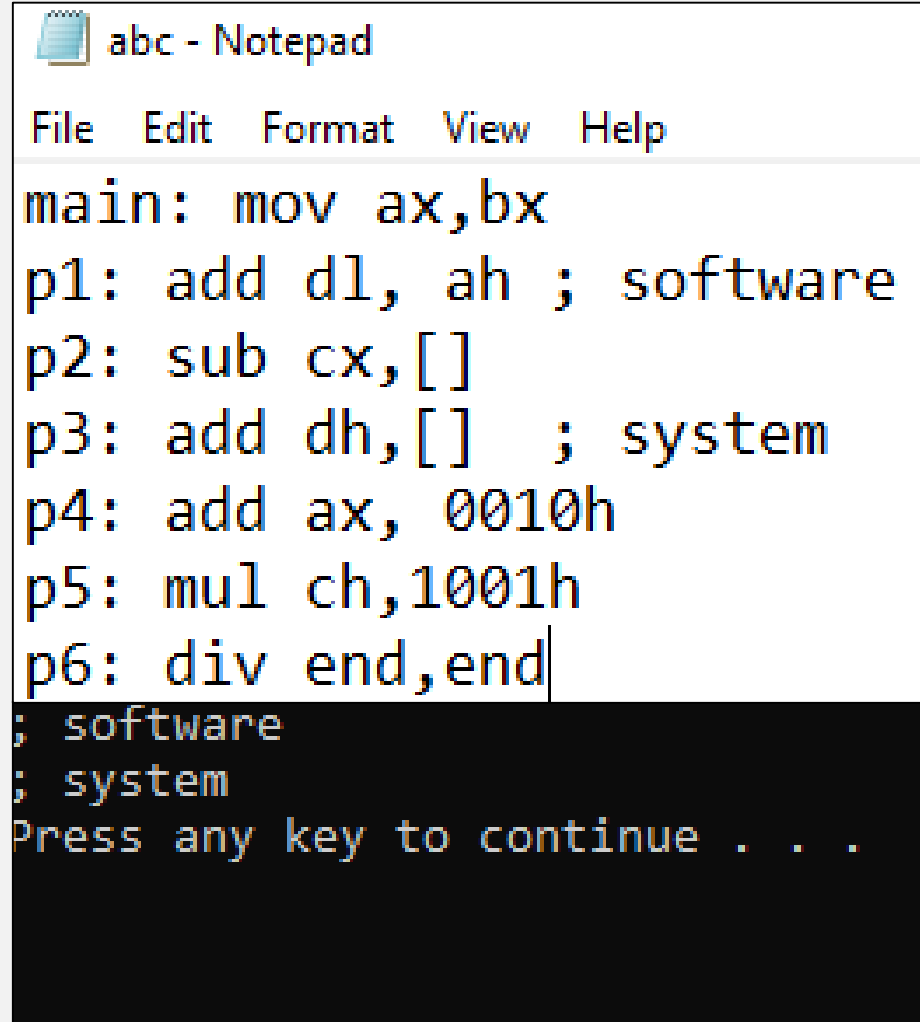
- Spotting up Labels.
- Recognizing the comments lines.
- Calculating the addresses.
- Splitting up R16 & R8 registers before or after the comma.



```
www.txt - Notepad
File Edit Format View Help
[main: mov ax,bx
p1: add dl,ah ; software
p2: sub cx,[]
p3: add dh,[] ; system
p4: add ax,0010h
p5: mul ch,1001h
p6: div end,end
```

SPOTTING UP COMMENTS

EX:01 Get the same output as shown below:



```
abc - Notepad
File Edit Format View Help
main: mov ax,bx
p1: add dl, ah ; software
p2: sub cx,[]
p3: add dh,[] ; system
p4: add ax, 0010h
p5: mul ch,1001h
p6: div end,end
; software
; system
Press any key to continue . . .
```

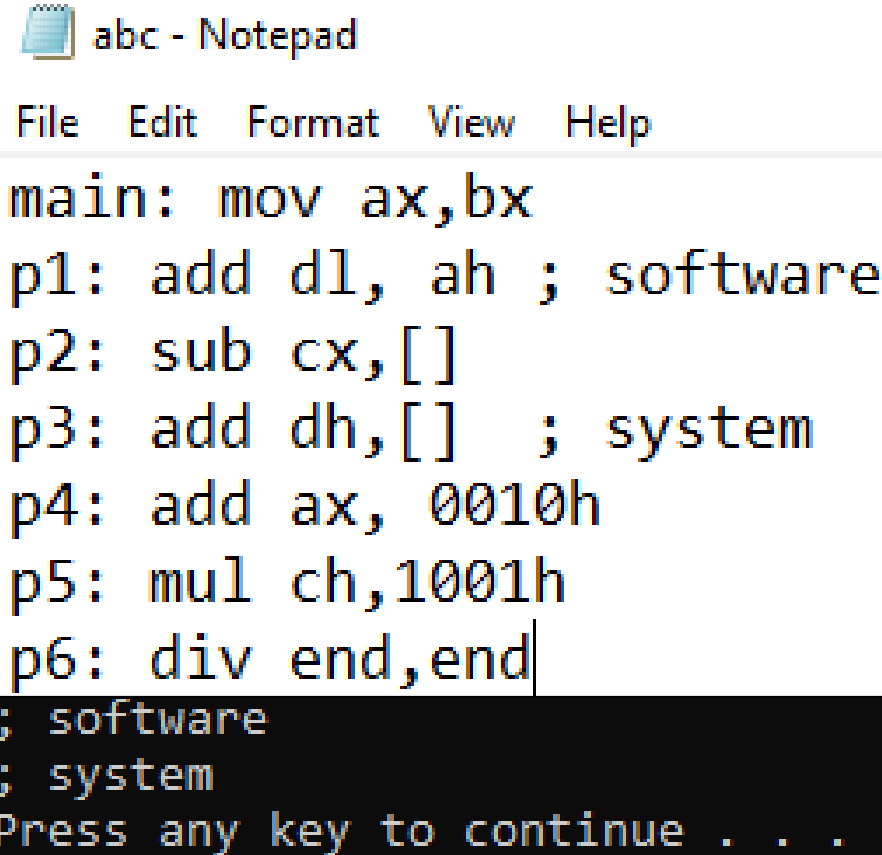
```

#include<iostream>
#include<string>
using namespace std;
void main(){
    char comment[100][100],str[100];
    int h = 0,z=0, j=0, i=0;
    FILE* fp;
    fp = fopen("abc.txt", "r+");
    if (fp == NULL)
    {
        cout << "Error in file!"; exit(0);
    }
    while (!feof(fp))
    {
        fgets(str, 100, fp);
        for(int i=0;str[i]!='\0';i++)
        {
            if (str[i] == ';')
            {
                for(z=i;z<strlen(str); z++)
                comment[h][z-i] = str[z];
                comment[h][z-i]='\0';
                cout<<comment[h];
                h++;
            }
        }
        fclose(fp);
        system("pause");
    }
}

```

SPOTTING UP COMMENTS

Get the same output as shown below:



```

abc - Notepad
File Edit Format View Help
main: mov ax,bx
p1: add dl, ah ; software
p2: sub cx,[]
p3: add dh,[] ; system
p4: add ax, 0010h
p5: mul ch,1001h
p6: div end,end
; software
; system
Press any key to continue . . .

```

SPOTTING UP COMMENTS

It's your turn for coding to have similar output!!



abc - Notepad

File Edit Format View Help

```
main: mov ax,bx
p1:  add dl, ah ; software
p2:  sub cx,[]
p3:  add dh,[] ; system
p4:  add ax, 0010h
p5:  mul ch,1001h
p6:  div end,end
```

```
main:
```

```
p1:
```

```
p2:
```

```
p3:
```

```
p4:
```

```
p5:
```

```
p6:
```

```
-----
The comments are:
```

```
; software
```

```
; system
-----
```

```
Press any key to continue . . .
```

```
#include<iostream>
using namespace std;
void main()
{
    FILE *fp;
    int k=0,j=0,i=0,h=0, z=0,m;
    char s[100],str[100],lable[100][100],comment[100][100];
    fp=fopen("abc.txt","r+");
    if(fp==NULL)
    {
        cout<<"Error in file ";
        exit(0);}
    while(!feof(fp))
    {
        fgets(s,100,fp);
        for(i=0;s[i]!='\0';i++)
        if(s[i]==';')
        {
            for( j=0;j<=i;j++)
            lable[k][j]=s[j];
            lable[k][j]='\0';
            cout<<lable[k];
            cout<<endl;
            k++;
        }
    }
    cout<< "-----" <<'\n'
    << " The comments are: " <<'\n';
    // it is important to set the file position indicator to the beginning of the given file stream
    rewind(fp);
    while (!feof(fp))
    {
        fgets(str, 100, fp);
        for(int i=0;str[i]!='\0';i++)
        {
            if (str[i] == ';')
            {
                for(z=i;z<strlen(str); z++)
                comment[h][z-i] = str[z];
                comment[h][z-i]='\0';
                cout<<comment[h];
                h++;
            }
        }
    }
    cout<< "-----" <<'\n';
    fclose(fp);
    system("pause");
}
```

THE END