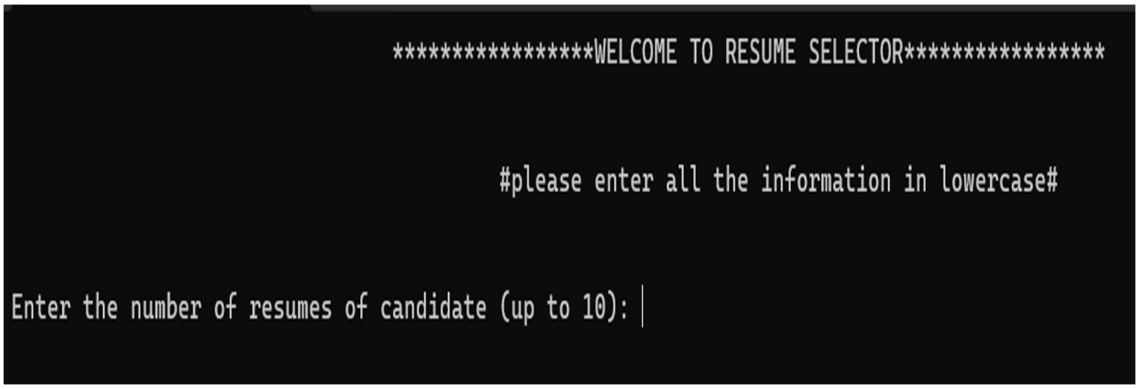
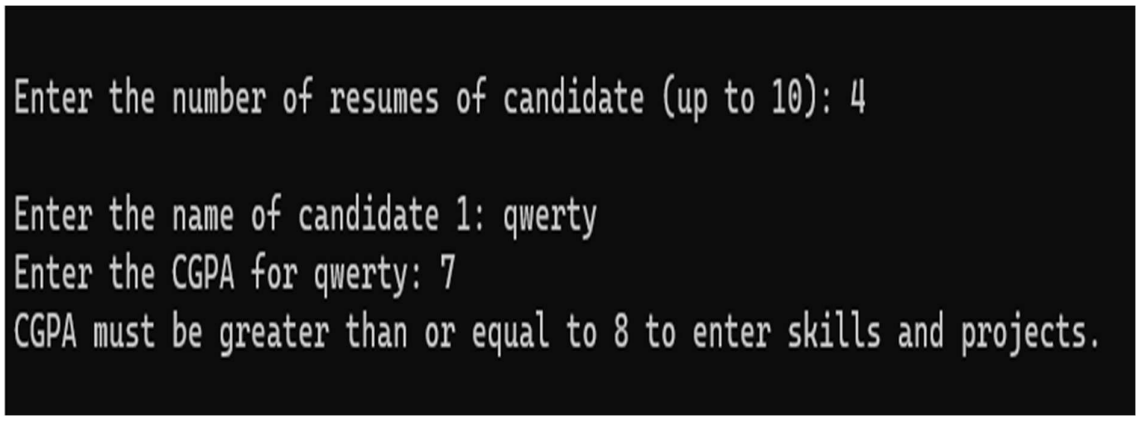
OUTPUT

The output of the provided code includes several key components that assist recruiters in evaluating and selecting candidates based on their resumes. Here's a brief overview of the expected output:



Welcome Message: The program begins with a welcoming message, indicating that it's a "Resume Selector" program. It also gives a prompt to the user to enter all the data in lowercase only.

User Input Prompts: Recruiters are prompted to input the number of resumes they want to evaluate and details for each candidate, including their name, CGPA, skills, and project experiences.



- Here we can see that if the candidate's cgpa is less than 8, then no further details are asked

```
Enter the name of candidate 2: asdf
Enter the CGPA for asdf: 8

Enter the number of skills for asdf (up to 10): 3
Enter unique skills (one per line) for asdf:
Skill 1: dsa
Skill 2: dsa
Skill 'dsa' is already listed.
Skill 2: os
Skill 3: c
```

```
Enter the number of projects for asdf (up to 5): 2

Enter the details of project 1 for asdf:
Project Name: ai
Project Domain: ai

Enter the details of project 2 for asdf:
Project Name: ai
Project 'ai' is already listed.
Project Name: c
Project Domain: c
Domain not matching with predefined domains.
```

- At times there might be a possibility that the recruiter might forget the skills/projects they have entered for a candidate and they might enter the same skill/project again, so to avoid that, the code is designed such that the recruiter can't enter the same skill/project again, it gives a suitable message
- Another prompt also occurs "Domain not matching with predefined domains.", it occurs when the project domain entered for the candidate is not needed/the company (recruiter) is not interested in hiring for that particular domain
- The interested domains are mentioned in the code under the function - *void addProjects(struct Resume* resume, const struct JobRequirements* job)* (This can be modified and varies from one company to another , in the above code the interested domain are : dsa, ai, cn, os, web)

Similarly the details of the rest of the candidates are entered as shown below

(Here for other 2 candidates)

```
Enter the name of candidate 3: zxcv
Enter the CGPA for zxcv: 8.5

Enter the number of skills for zxcv (up to 10): 3
Enter unique skills (one per line) for zxcv:
Skill 1: dsa
Skill 2: c
Skill 3: cn

Enter the number of projects for zxcv (up to 5): 2

Enter the details of project 1 for zxcv:
Project Name: ai
Project Domain: ai

Enter the details of project 2 for zxcv:
Project Name: os
Project Domain: os

Enter the name of candidate 4: plmokn
Enter the CGPA for plmokn: 9

Enter the number of skills for plmokn (up to 10): 4
Enter unique skills (one per line) for plmokn:
Skill 1: dsa
Skill 2: os
Skill 3: c
Skill 4: cn

Enter the number of projects for plmokn (up to 5): 3

Enter the details of project 1 for plmokn:
Project Name: dsa
Project Domain: dsa
```

```
Enter the details of project 2 for plmokn:
Project Name: cn
Project Domain: cn

Enter the details of project 3 for plmokn:
Project Name: ai
Project Domain: ai
```

Ranked Resumes: After entering all candidate details, the program ranks the resumes based on their total scores, which are calculated using a combination of skill scores and project scores. The ranked resumes are displayed, providing recruiters with a sorted list of candidates from highest to lowest total score.

- The ranking is done:

    I. For Skills : based on the score assigned to the individual skill in the above code *(here, dsa=50 ,ai=40,cn=30,os=20 and any other skill=0)*.
    II. For Projects : based on the domain in which the candidate has done their projects (*here,if domain is : dsa=60 ,ai=50,cn=40,os=30, web=20 and any other domain=0)*.

This can be modified and varies from company/role which hiring is done for.

Based on scores been assigned to the skills/projects total skill/project score is been calculated and displayed.

```
Ranked Resumes:

Name: plmokn
CGPA: 9.00

Skills:dsa os c cn
Projects: dsa (dsa) cn (cn) ai (ai)

Skill Scores:
dsa: 50
os: 20
c: 0
cn: 30

Project Scores:
dsa (Domain: dsa): 60
cn (Domain: cn): 40
ai (Domain: ai): 50

Total Skill Score: 100
Total Project Score: 150


Name: zxcv
CGPA: 8.50

Skills:dsa c cn
Projects: ai (ai) os (os)

Skill Scores:
dsa: 50
```

```
Skill Scores:
dsa: 50
c: 0
cn: 30

Project Scores:
ai (Domain: ai): 50
os (Domain: os): 30

Total Skill Score: 80
Total Project Score: 80


Name: asdf
CGPA: 8.00

Skills:dsa os c
Projects: ai (ai) c (c)

Skill Scores:
dsa: 50
os: 20
c: 0

Project Scores:
ai (Domain: ai): 50
c (Domain: c): 0

Total Skill Score: 70
Total Project Score: 50
```

Search Menu: Recruiters are then presented with a search menu, where they can choose to search for resumes by name or skill. Upon selecting a search option, the program prompts for the relevant search criteria and displays the matching resumes along with their details.

```
Search Menu:
1. Search by Name
2. Search by Skill
3. Exit
Enter your choice: 1
Enter the name to search: qwerty
Resume with name 'qwerty' not found.

Search Menu:
1. Search by Name
2. Search by Skill
3. Exit
Enter your choice: 1
Enter the name to search: asdf
Resume found:
Name: asdf, Skills: dsa os c Total Score: 120
```

```
Search Menu:
1. Search by Name
2. Search by Skill
3. Exit
Enter your choice: 2

Enter the skill to search: java
No resumes found with skill 'java'.

Search Menu:
1. Search by Name
2. Search by Skill
3. Exit
Enter your choice: 2

Enter the skill to search: dsa

Resumes found with skill 'dsa':
Name: plmokn, Skills: dsa os c cn Total Score: 250
Name: zxcv, Skills: dsa c cn Total Score: 160
Name: asdf, Skills: dsa os c Total Score: 120
```

```
Search Menu:
1. Search by Name
2. Search by Skill
3. Exit
Enter your choice: 3

Process returned 0 (0x0)   execution time : 340.140
 s
Press any key to continue.
```

Suitable messages are displayed if name/skills not found and program is exited on entering number 3.

Record File: Additionally, the program writes the ranked resumes and their details to a file named "record.txt" for future reference or documentation purposes.

(Details of the candidates as shown is stored one below the other)

```
Name: plmokn
CGPA: 9.00

Skills:dsa os c cn
Projects: dsa (dsa) cn (cn) ai (ai)

Skill Scores:
dsa: 50
os: 20
c: 0
cn: 30

Project Scores:
dsa (Domain: dsa): 60
cn (Domain: cn): 40
ai (Domain: ai): 50

Total Skill Score: 100
Total Project Score: 150
```

```
Name: zxcv
CGPA: 8.50

Skills:dsa c cn
Projects: ai (ai) os (os)

Skill Scores:
dsa: 50
c: 0
cn: 30

Project Scores:
ai (Domain: ai): 50
os (Domain: os): 30

Total Skill Score: 80
Total Project Score: 80
```

```
Name: asdf
CGPA: 8.00

Skills:dsa os c
Projects: ai (ai) c (c)

Skill Scores:
dsa: 50
os: 20
c: 0

Project Scores:
ai (Domain: ai): 50
c (Domain: c): 0

Total Skill Score: 70
Total Project Score: 50
```

Overall, the output of the code provides recruiters with a comprehensive overview of candidate qualifications, ranked based on their suitability for specific job requirements. It offers tools for efficient evaluation, comparison, and search, empowering recruiters to make informed decisions in their hiring process.