ACTIVITY 2: A C-PROGRAM TO CHECK ATHLETE QUALIFICATION BASED ON PERFORMANCE

Name- Swasti S. Mahadeokar
Roll no- 53
Branch- Computer Science
Class- CE2

1) RESEARCH

1) In the real life, athletes go through qualification stages before reaching the finals.
2) Typical events could include swimming, sprint (100m race), long jump and shooting.
3) These stages include: fitness checks, national trials, international qualifiers, semifinals, and finals.
4) Only athletes who succeed at every stage can win medals; failure at any stage leads to elimination.
5) Performance is evaluated through scores, timings, or ranks, which can be mapped to a grading system (Gold, Silver, Bronze, Qualified, Disqualified).

2) ANALYSIS

1) This system is sequential: you cannot skip a stage.
2) It can be coded using conditional checks (if-else) where each stage must be cleared to move to the next.
3) Error handling is needed for invalid inputs (e.g., negative scores, or out-of-range performance values).
4) The project resembles real-world performance evaluation but is modeled in a gamified way with medals as outputs.
5) The purpose of this project is to simulate an athlete’s qualification journey across multiple sports trials and

to demonstrate how conditional checks + averages can evaluate performance in programming.

3) IDEATE

1) Input: The program will ask for athlete's scores in these events:
a) Swimming
b) Sprint (100m race)
c) Long Jump
d) Shooting

2) Logic:
a) If score in any event < 40 - Athlete is "Disqualified" .
b) If in all events, score ≥ 40 - calculate average score.

3) Grading / Medals:
a) Gold Medal - Avg ≥ 90
b) Silver Medal - 75–89
c) Bronze Medal - 60–74
d) Qualified Only (No Medal) - 40–59
e) Disqualified - Fail in any event

4) Output will display the score of each trial event, whether athlete passed/failed each stage and print the final medal or status at the end.
5) References:
https://worldathletics.org/competition/qualifying-standards
https://olympics.com/ioc/athlete-eligibility

4) BUILD

```
#include <stdio.h>

void main() {
```

```
    int swimming,sprint,long_jump,shooting;
    int average;

    printf("Enter score in swimming:\n");
    scanf("%d", & swimming);

    printf("Enter score in sprint race:\n");
    scanf("%d", & sprint);

    printf("Enter score in long jump:\n");
    scanf("%d", & long_jump);

    printf("Enter score in shooting:\n");
    scanf("%d" , & shooting);

    if (swimming<40 || sprint<40 || long_jump<40 || shooting<40 ) {
        printf("Athlete is Disqualified!\n") ;

    } else {
        average= (swimming+ sprint+ long_jump+ shooting)/4;

        printf("Athlete qualified all the events!\n");
        printf("Average score is= %d\n", average);}


       if (average>=90) {
           printf("Result:The candidate has been classified as an elite with
a gold medal!\n"); }

        else if (average>= 75) {
            printf("Result:The candidate has passed the event with silver
medal!\n");}
```

```
        else if (average>= 60) {
            printf("Result: The candidate has passed the event with bronze
medal!\n ");}

        else if (average>= 40) {
            printf(" The candidate has barely passed the event with
participation certificate!\n");}



        }
```

5) TESTING

Test 1: Athlete is disqualified
Input:
Swimming- 60
Sprint race- 50
Long jump- 30
Shooting- 55
Output: Athlete is disqualified!

Test 2 : Elite and Gold medal
Input:
Swimming- 90
Sprint race- 95
Long jump- 98
Shooting- 96
Output: Athlete has been classified as an elite with gold medal !

Test 3: Silver medal
Input:
Swimming- 80

Sprint race- 85
Long jump- 70
Shooting- 75
Output: Athlete has passed the event with silver medal !

Test 4: Bronze medal
Input:
Swimming- 60
Sprint race- 65
Long jump-66
Shooting- 68
Output: Athlete has passed the event with bronze medal !

Test 5: Barely passed
Input:
Swimming - 50
Sprint race- 45
Long jump- 44
Shooting- 48
Output: Athlete has barely passed the event with participation certificate !

5) CONCLUSION

This project successfully demonstrates the events of the competition of the athletes and calculates average scores and displays the corresponding results on the output screen.