

Requirements (25 pts)

| | 5 | 3 | 1 |
|-----------------|---|---|---|
| The “what” part | Requirements thoroughly explain what the program will and will not do | Requirements explain what the program will and will not do but some detail is lacking. | Very little detail in your requirements. |
| Presentation | Requirements are presented in a clear, concise method that is easy to understand | Requirements are presented in a clear, concise method that is easy to understand for the most part but occasionally something is hard to understand | Requirements are presented in a in a messy that sounds like you really don’t understand what you are talking about or there is no real plan |
| Thoroughness | I could hand your requirements to anyone and they could start designing it | I could hand your requirements to anyone and they could start designing it with a few questions | I could hand your requirements to anyone and they wouldn’t know where to start |
| Error Checking | The requirements discuss error checking detailing how you will handle improper input for multiple sections, etc | The requirements discuss error checking detailing how you will handle improper input for at least 1 section | What error checking? |
| Input | You discuss what types of input the user will give (i.e. grades will be entered in by percentage) for multiple sections | You discuss what types of input the user will give (i.e. grades will be entered in by percentage) for 1 section | Not discussed |

Design - (30 pts)

| | 5 | 3 | 1 |
|-----------------|---|------------------------------------|---------------------------------------|
| Data Structures | Discusses on a detailed about what data | Mentions how the program will flow | Missing a lot of detail. Doesn’t talk |

| | | | |
|----------------|---|---|---|
| | structures and classes will be used and how the program will flow | on a high level, but lacks in some detail | about at least 1 of the subjects |
| Presentation | Design is presented in a clear, concise method that is easy to understand either via paper | Design is presented in a clear, concise method that is easy to understand but sometimes things are hard to understand | Design is presented in a in a messy format and sounds like you really don't understand what they are talking about or there is no real plan |
| Thoroughness | I could hand your design to anyone and they could start coding it | I could hand your design to anyone and they could start coding it with a few questions | I could hand your design to anyone and they wouldn't know where to start |
| Input | Discusses what the user is expected to input and how it will be handled in detail | Discusses what the user will input at a high level | Very little discussion |
| output | Discusses what the output will look like in detail | Discusses output at a high level | Very little discussion |
| Error checking | Discusses what the error checking will look like in detail - how will it be handled, what will you check for, etc | Discusses error checking at a high level | Very little discussion |

Code & Presentation (60 pts)

| | 5 | 3 | 1-0 |
|------------|---|---|---|
| Attendance | Your program has a way of entering attendance that makes sense and is usable. It is demonstrated. | Your program can take attendance, but the method doesn't make sense or it is hard to use. It is demonstrated. | Missing or not working |
| Java use | You have used what you learned in class and have made good decisions about | You have made use of what you have learned in class and used what was | There are obvious gaps in what you chose to use to solve your |

| | | | |
|-------------------------|---|--|--|
| | the best way to solve the problems presented. | necessary to solve the problem but some of the decisions you made are questionable (i.e. no loops for something that would be naturally solved by loops) | problem or code is incomplete. |
| Aesthetics | Program is well commented – all classes and methods are commented with comments that make sense | There are some comments and they make sense | There is 1 comment or less |
| Variables | Variable Names are descriptive throughout the program | Variable Names are descriptive throughout the program with a few exceptions | Variable Names are descriptive < 50% of the time |
| Methods | Methods are used throughout the program to break up all logical sections | There are methods in the program but there are very few or this is no logic with them | There is 1 method or less |
| Registering for classes | Your program has a way of registering for classes that makes sense and is usable. It is demonstrated. | Your program can register for classes but the method doesn't make sense or it is hard to use. It is demonstrated. | Missing or not working |
| Entering grades | Your program has a way of entering grades that makes sense and is usable. It is demonstrated. | Your program can enter grades but the method doesn't make sense or it is hard to use. It is demonstrated. | Missing or not working |
| Classes | Classes are used throughout the program to break up all logical sections | | You don't use classes |

| | | | |
|-----------|---|---|---|
| Usability | Your program follows a logical flow and is easy to figure out | I can figure out how to run it but things are hard to get to or figure out | I need a manual to run your program |
| Input | There are nice prompts telling me what to enter when | There are prompts but I'm guessing what to enter | What prompts |
| Output | All output is formatted nicely in that you tell me what I'm seeing and why | You output things but you are missing the explanation | Some things aren't output |
| Video | There is a video showing all aspects of your program working. You explain what your program is doing. The video is clear. | There is a video showing your program working. You either don't show all aspects or there isn't a lot of explanation. The video is clear most of the time | Poor quality video, missing a lot or not present. |