

2020 Programming

Rule Book

# Objective

The goal of the Programming category is to encourage engineering students to produce a piece of industry-quality software with all of the proper user and administrative documents. The teams will use their software development skills, their technical writing abilities, and their project management skills to design a solution to a posed problem. This solution will then be presented to company executives (judging panel).

# Competition Leads

**Sanjay Kumar Jenita John**

**(VP Competition) (Programming Lead)**

# General rules

* All questions regarding the competition problem must be asked during the welcome and briefing session. No questions will be answered during the design and build stage.
* Laptops/tablets and reliable internet are required.
* There will be approximately 8 hours and due by 2 am for the build stage to complete the design. Finished or not, the design must be submitted on time or will not be accepted.
* You will be allowed to leave before the allotted time.
* The presentation and any presentation materials can be done in either English or French but must be consistent in language.
* **All additional required files and libraries must be provided along with all source code within a single zip file and handed in by the deadline on Saturday, Nov 28 at 2:00 am to the REC Committee. NO LATE SUBMISSIONS WILL BE ACCEPTED!**
* The report and presentation must be submitted in a PDF format by the specified deadline and cannot be modified after submission. It is the team’s responsibility to do this before the end of the build time.

# Slack and Zoom Rules:

* Any competition issues should be addressed publicly on Slack. For programming related concerns mention the above leads names in the <channel-name> with your messages if necessary. (Ex: @Jenita John <your-message>)
* Cameras must be on throughout the competition.
* All the team members must share their screen during the design phase.
* Only the presenters’ screen needs to be shared during the presentation period. (Recommendation: Assign one member to share ppt and one member to share the code)

# General Guidelines

## Judging

Each team will be judged based on the developed solution to the given challenge, the submitted report as well as their presentation. Applications will be judged based on functionality and overall quality of code. Reports will be judged based on the team’s technical writing quality and explanation of code methodology. The presentation will be judged based on design justification and team participation.

## Software Resources

* There are no restrictions on what languages used for the competition.
* All information used by competitors must be referenced very carefully.
* Competitors are not permitted to submit work completed by anyone other than the members of their team.
* If competitors decide to recycle their own or someone else’s code it must be clearly cited in the presentation. In addition, the competitors also need to clearly explain why and where the recycled code was used in their software.
* The judges hold the right to ask any team member to describe what a particular section of the code does at any given point during the presentation.
* If there is any evidence that competitors are submitting plagiarized work, the entire team will be eliminated from the competition.

## Timekeeping

***During the design phase***

* The design phase starts at **6 pm on Friday, November, 27**.
* The time remaining in the competition will be announced three (3) hours, one (1) hour, thirty (30) minutes and ten (10) minutes before the deadline

***During the presentation***

* Your presentation time will be provided to you on the day of the competition.
* Time is halted when a judge asks a question during the presentation
* Time is halted when a team member answers a question asked by a judge
* The remaining will be indicated to the competitors 10 minutes, 5 minutes and 1 minute before the end of the allotted time for the presentation to the judges

# Competition Deliverables:

Teams in Programming are required to design, develop, test, provide documentation, and construct a presentation of their project during the limited time provided in the first phase of the competition and then present an oral presentation and demonstrate their solution in the second phase of the competition. All deliverables must be submitted electronically before the end of the design period. This includes the code which will be used to evaluate the team’s solution, their presentation, and report. Only the code provided to the Programming Director will be used for evaluation.

The oral presentation should shortly summarize the team’s layout, the design process, the management process, and the development process. If there were required components that could not be constructed in the time given, teams need to highlight the mistakes made and provide an explanation on how the problem could be solved in the future. If the solution included any open-source libraries, the presentation should highlight the components that contain the code and if there is an alternative library that should have been used. Teams should contain a separate, larger portion of the presentation that showcases the solution for approval. The presentation should introduce the software, present the core functions of the software, how the program’s components work from a development standpoint, its user documents, the installation method, and any unique components of the solution that were not suggested in the problem. The presentation should also include who the potential customers may be, what benefits the program would give this customer, and provide a short demo of the product and its components for the panel. Judges reserve the right to ask questions during the presentation. Teams must deliver answers within a reasonable time (determined by event official) to avoid deduction and/or a committee investigation (based on the context of the question).

## *Outfiles*

Include in the submission folder the output CSV file your program creates as it analyzes the input.

Format: [team name] - output.csv

## *Readme.txt*

Include in the submission folder readme.txt, with the following information:

* Administrative information such as names, emails of team members, group number and project title
* Location of all code used for the project with directories
* Instructions to compile and run the project for the judges

## *Report*

Include in the submission folder a brief, 2-page report outlining the engineering problem to be solved:

* Identify the stakeholders and the problem, explaining why the problem is relevant to stakeholders
* Identify the chosen solution and how it addresses the chosen problem
* Identify the target audience
* Give a high-level overview of the solution design
* Shortly summarize the design process, management process, and development process
* If there were required components that could not be constructed in the time given, highlight the mistakes made and provide an explanation on how the program could be solved in the future
* If the solution included any open-source libraries, highlight the components that contain the code and if there is an alternative library that should have been used
* Indicate the core functions of the software and how the program’s components work from a development perspective
* Clearly mention the installation method, and any unique components of the solution that were not suggested in the problem.

Format: [team name] - report.doc

## *Presentation*

Presentation order shall be determined randomly and announced 30 minutes before the presentations commence. Teams are not allowed to switch places in the presentation order. Include in the submission folder your final presentation for the judges demonstrating the solution and how it works. Oral presentations should:

* Be no more 12 minutes in length, NOT including 8-minute Q & A period
* Briefly summarize the contents of the report
* Walk the judges through the implementation, including technologies used and any difficult technical challenges the team has faced
* Demonstrate the solution, including inputs defined by judges and/or test cases
* Judges can ask questions at any time during the presentation. The clock will be stopped during these interruptions
* The competitors cannot include any material in their presentation which is not included within their submitted written reports or presentation materials.

Format: [team name] - presentation.ppt

# Submission:

A complete submission consists of:

* A PDF of your powerpoint/report for your presentations
* Any codes, you must be put in a .ZIP file.

All submissions should be sent to **rec.competitions@ress.ca** by the submission deadline @2:00 A.M, Saturday, November 28th, 2020. All submissions must be submitted through a ryerson email. The subject line in the email should clearly indicate the following:

* Competition name, team name, one student’s name and last name.

*For example:*

***Communication - Team Alpha - Sanjay Kumar***

# Programming Rubric

**Criteria Points Design & Performance**

## Strategy / 60

**Algorithm / 25**

Does the algorithm/code achieve the desired outcome?

Ingenuine approach

Does the solution address the proper audience?

**Performance/Code / 25**

Code efficiency

Structure

Readability

Execution/Application

Does the program work and execute properly?

User interactivity

Framework layout and visuals

Content is relevant to the given problem

A core relevant feature is implemented

**Originality/Resource Management / 10**

Memory Usage Efficiency

Program’s CPU Usage

Idea/Concept is new and original

All group members equally contributed in every stage

**Bonus**

Unit Test /5

Additional Features /5

## Report / 20

### Structure and Content

Was it clear that team members knew what the problem was? Was the planning of the solution clear? Did all members contribute to the problem-solving process? Was the team communication clear? Such as: Proper use of jargon? Not extremely technical? Were visual aids effectively used? Such as: flow charts, UML diagrams, etc. - Were any important details overlooked? Analysis on your results.

### Code Methodology

Was the code architecture explained? - Were the benefits and principles of the design explained? - Was the program code made in a systematic method? - Were any open source components made apparent? - Was the work split up amongst group members in a logical way?

## Presentation /20

Design Process

Design Justification

Design Critique

Voice

Articulation and Timing

Visual Aids

Response to Questions

## Penalties

Plagiarism - 50

Insufficient Citation - 50

Documents Received After Deadline - 50

Absent Team Member - 25

**Total /100**