Albert Ratschinski (5154309)
Severin Plewe (5333060)

Exercise Sheet Nr. 1
(Deadline Wednesday, 01-05-2024, 15:59)

Task	1.1	1.2	2.1	2.2	2.3	2.4
Completed	☒	☒	☒	☒	☒	☒
Feedback	☒	☐	☒	☐	☐	☒

Exercise 1 - Cyclomatic Complexity

Exercise 1.1

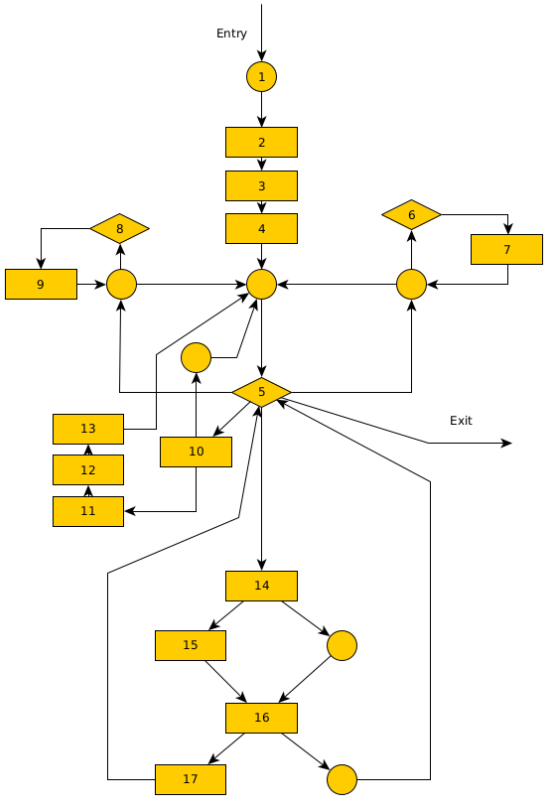

Abbildung 1: Control Flow Graph for *qucikSort()*

To calculate the cyclomatic complexity of the given control flow graph we can use the formula:

$$v(G) = |E| - |V| + p$$

We yeild:

Number of nodes: 23
Number of edges: 31
$V(G) = 31 - 23 + 2 = 6$

Exercise 1.2

The alternative choice of the CFG construction would not alter the value of the cyclomatic Complexity metric since:

$$V(G) = |E| - |V| + p = 10 - 9 + 2 = 3$$

which is the holds the same value as the original CFG. We can thereby conclude that the cyclomatic complexity metric is independent of wheather we chose one version or the other. For overview sake I would argue that represented version is way more readable and easier to understand.

Exercise 2 - Cost Estimation

Exercise 2.1

Estimation of Overall Development Effort in Person-Months (PM)

1. **Definition of Person-Month:** A person-month represents the amount of work one person can complete in one month. It's calculated by multiplying the number of people working on a project by the number of months they work.
2. **Calculation:**
   - Each team member contributes approximately 180 hours of work.
   - Assuming an average of 6.5 team members.
   - Total hours contributed by the team per month: $6.5 \times 180 = 1170$ hours.
   - Typical number of working hours in a month is assumed to be 160.
   - Person-months = $\frac{1170 \text{ hours}}{160 \text{ hours}} \approx 7.3125$ PM.
3. **Result:** The estimated overall development effort for this project is approximately 7.31 person-months.

Question from the student: *How would we calculate the PM from given information of the slide?*

Definition of [PM (person-months)] from the slides:

Person-months (PM) is a unit of measurement that represents the amount of work performed by a person in one month.

$$E = a \cdot (\frac{S}{kDSI})^b$$

Where:

- E is the effort in person-months
- S is the size of the software product in DSI (Delivered Source Instructions)
- a and b are constants
- k is a constant *what does it stand for?*

Exercise 2.2

We conducted a thorough analysis of the requirements specified in Appendix A for the development of the video game. Taking into consideration the multifaceted nature of the functional and quality requirements, as well as the additional constraints delineated, we have estimated the delivered KLOC (Thousand Lines of Code) to fall within the range of 50-70 KLOC.

- Functional requirements complexity
- Quality requirements demands
- Additional constraints imposition
- Continuous code maintenance necessity
- Complexity of player interactions
- Real-time gameplay requirements
- Scalability and extensibility considerations
- Usability and user experience implementation

Exercise 2.3

Cost Driver	**Rating**
Product Attributes	
Required Software Reliability	High
Size of Application Database	High
Complexity of the Product	Nominal
Hardware Attributes	
Run-time Performance Constraints	High
Memory Constraints	Nominal
Volatility of the Virtual Machine Environment	Nominal
Computer Turnaround Time	Nominal
Personnel Attributes	
Analyst Capability	Nominal
Applications Experience	Nominal
Software Engineer Capability	Nominal
Virtual Machine Experience	Nominal
Programming Language Experience	Nominal
Project Attributes	
Use of Modern Programming Practices	Nominal
Use of Software Tools	Nominal
Required Development Schedule	Nominal

Tabelle 1: Cost Drivers and Ratings

- Required Development Schedule **Rating:** Nominal
  Explanation: The project's development schedule is expected to follow the nominal timeline, with minor flexibility but no significant deviations.

- Required Software Reliability **Rating:** High
  Explanation: Given the potential consequences of software failures in the game project, a high level of reliability is crucial to avoid significant financial losses or human inconveniences.

- Use of Modern Programming Practices **Rating:** Nominal
  Explanation: While the project aims to incorporate modern programming practices, such as structured design and incremental development, not all practices are fully leveraged, resulting in a nominal rating.

Exercise 2.4

Calculation of Effort Adjustment Factor (EAF)

$$\begin{aligned} EAF &= \prod_{i=1}^{n} C_i \\ &= 1.15 * 1.08 * 1.00 * 1.11 * 1.00 * 1.00 * 1.00 * 1.00 * 1.00 * 1.00 * 1.00 * 1.00 * 1.00 * 1.00 * 1.00 \\ &= 1.3786 \end{aligned}$$

Calculation of estimated effort in person-months (E)

$$\begin{aligned} E &= a * (KLOC)^b * EAF \\ &= 3.2 * (50)^{1.05} * 1.3786 \\ &= 268.2296 \end{aligned}$$

To break it down:

- Each team member contributes approximately 268 hours of work.
- Assuming an average of 6.5 team members.
- Total hours contributed by the team per month: $6.5 \times 268 = 1742$ hours.
- Typical number of working hours in a month is assumed to be 160.
- Person-months $= \frac{1742 \text{ hours}}{160 \text{ hours}} \approx 10.8875$ PM.
- Therefore the difference between the estimated effort and the actual effort is $10.8875 - 7.3125 = 3.575$ PM.

Reasons for the deviation in the estimated effort:

- Inaccurate Estimation of Cost Drivers: The ratings assigned to the cost drivers in the COCOMO calculation might not accurately reflect the true impact of these factors on the project's effort.
- The COCOMO calculation might have considered additional factors or complexities that were not fully accounted for in the initial estimation.