



COURSEWORK ON SYSTEM PROGRAMMING

Syllabus

Requisites of the Course

| | |
|----------------------------------|--|
| Cycle of Higher Education | <i>First cycle of higher education (Bachelor's degree)</i> |
| Field of Study | <i>12 Information Technologies</i> |
| Speciality | <i>123 Computer Engineering</i> |
| Education Program | <i>Computer Systems and Networks</i> |
| Type of Course | <i>Normative</i> |
| Mode of Studies | <i>full-time</i> |
| Year of studies, semester | <i>3 year (5 semester)</i> |
| ECTS workload | <i>1 credit (30 hours of self-study)</i> |
| Testing and assessment | <i>Coursework / defense of Coursework</i> |
| Course Schedule | |
| Language of Instruction | <i>English</i> |
| Course Instructors | <i>PhD, Associate Professor, Valerii Pavlov, pavlovvg@ukr.net</i> |
| Access to the course | <i>https://campus.kpi.ua/tutor/index.php?mode=mob&show&irid=218710</i> |

Outline of the Course

1. Course description, goals, objectives, and learning outcomes

*The academic discipline "Coursework on System Programming" belongs to the mandatory (regulatory) educational components of the education program, namely to the professional training cycle. It has the code **PM 16** in the list of components of the educational program.*

***Reason and motivations for studying:** the need to study the academic discipline «Coursework on System Programming» is to consolidate, deepen, generalize theoretical knowledge and practical skills, which students receive during the study of the discipline «System Programming», understanding of the principles of software control of the processor itself.*

***The goal of the "Coursework on System Programming" course is:** Obtaining skills in the development of system programs in accordance with the requirements of the specification or technical requirement and the preparation of a set of documents for programs for their customer or users. Study of the basic requirements for the basic documents of system programs and obtaining skills in the design of such documents on examples of the implementation of system programs.*

***The purpose of the discipline** is the formation of a number of competencies among students, namely:*

CAPACITY:

- to understand the principles of processor control at the software level;*
- to understand the sequence of actions during the development of a program in machine language;*
- to analyze the structure of the program in machine language;*
- to develop the programs on Assembler language;*
- to analyze the processes that are carried out during the compilation and linking of programs;*
- to use macro-assemblers opportunities in programming;*

- to analyze the use of computer system resources by programs, in particular memory;
- also competencies GC-3, GS-7, PC-2, PC-11, PC-12, PC-13, PC-15 and PC-18 from the MATRIX 5 of EDUCATIONAL PROFESSIONAL PROGRAM.

1.2. The main tasks of the academic discipline.

After mastering the academic discipline, students must demonstrate such learning outcomes:

KNOWLEDGE:

- structures and components of System Software;
- machine language command structures and formats;
- basic storage formats of data according to IEEE 754 -2008;
- x86-64 processor modes;
- Assembler program structures;
- Assembler compiler directives;
- interaction of system programs during their execution.

ABILITY:

- to solve the problems of analysis and synthesis of means typical for the specialty;
- to think systematically and use creative abilities to form new ideas;
- to develop Software for Embedded and Distributed Applications, Mobile and Hybrid Systems; calculate, operate typical equipment for the specialty;
- to search for information in different sources to solve problems of Computer Engineering;
- to work effectively both individually and as part of a team;
- to combine Theory and Practice, as well as make decisions and develop a Strategy of actions to solve Problems of the Specialty taking into account Universal Values, Social, National and Production Interests;
- to perform experimental research on professional topics;
- to evaluate the results obtained and defend the decisions made reasonably;
- to create and maintain Databases.

SKILLS:

- to experiment, collect data and simulation in Computer Systems;
- to communicate verbally and in writing on professional matters in Ukrainian and at least one of the official EU languages (English, Germany, Italian, French, Spanish etc.);
- to adapt to new situations, to justify, make and implement decisions within the competence.

2. Prerequisites and post-requisites of the course (the place of the course in the scheme of studies in accordance with curriculum)

Interdisciplinary Connections: To successfully study the «Coursework on System Programming» students must master the material and have certain knowledge, skills and abilities in such disciplines:

- GM9 - «Programming»,
- PM1 - «Computer Logic»,
- PM6 - «Computer Architecture»,
- PM13 - «Algorithms and Methods of Computation»,
- PM7- «System Programming».

Knowledge and skills acquired during the study of the discipline «Coursework on System Programming» can be used in the future when mastering the following courses:

- PM6 – «Computer Architecture»,
- PM9 - «System software»,

- PM17 – «Course work on Computer Architecture»,
- PM18 – «Course work on System software»,
as well as during diploma design.

3. Content of the course

Coursework is carried out on an individual task and is prepared for defense in the final period of theoretical training. Coursework must be prepared for defense within the period stipulated in the assignment and agreed with the teacher. To the defense of the Coursework appears explanatory note in the composition of:

- title page;
- job sheet;
- abstract;
- index;
- introduction;
- the main part, includes sections:
 - problem statement;
 - description of methods of solving the problem;
 - planning experiments;
 - description of the software product;
 - results of the research on control examples;
- conclusions;
- references.

3.1 Stages of the Coursework

Main stages of the course work:

- analysis of the subject area relative to the task;
- development of an algorithm for the functioning of the program;
- program design and coding;
- program testing;
- design of the explanatory note;
- defense of Coursework.

4. Bibliography

4.1. Basic:

1. Pavlov Valerii. Methodical instructions for the implementation of coursework for students of the specialty 123 «Computer Engineering»: <https://campus.kpi.ua/tutor/index.php?mode=mob&show&irid=218710>
2. Abdulaziz Ghuloum. An Incremental Approach to Compiler Construction, Conference: Scheme and Functional Programming Workshop, 2006. URL: <http://scheme2006.cs.uchicago.edu/11-ghuloum.pdf>.
3. Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman. Compilers. Principles, Techniques, and Tools, – ADDISON-WESLEY, 1986, - 796 p.
4. Alfred V. Aho, Jeffrey D. Ullman. The theory of parsing, translation and compiling. Two-volume series. – Prentice-Hall, 1972, - 2051 p.
5. Wirth Niklaus. Compiler Construction. – ADDISON-WESLEY, 2000, - 132 p.

4.2 Additional

1. ГОСТ 19 001-77 ЕСПД. Общие положения.
2. ГОСТ 19.101-77 ЕСПД. Виды программ и программных документов.

3. ГОСТ 19.102-77 ЕСПД. Стадии разработки.
4. ГОСТ 19.103-77 ЕСПД. Обозначение программ и программных документов.
5. ГОСТ 19.105-78 ЕСПД. Общие требования к программным документам.
6. ГОСТ 19.106-78 ЕСПД. Требования к программным документам, выполненным печатным способом.
7. ГОСТ 19.202-78 ЕСПД. Спецификация. Требования к содержанию и оформлению.
8. ГОСТ 19.401-78 ЕСПД. Текст программы. Требования к содержанию и оформлению.
9. ГОСТ 19.402-78 ЕСПД. Описание программы.
10. ГОСТ 19.404-79 ЕСПД. Пояснительная записка. Требования к содержанию и оформлению.
11. ГОСТ 19.701-90 ЕСПД. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения.

Educational content

5. Methodology

The Coursework consists of seven stages, which are shown in the table 5.1.

Table 5.1

| Week of semester | Work stage name |
|-------------------------|--|
| 4 | <i>Getting a topic and assignment for Coursework</i> |
| 5 - 7 | <i>Analysis of the task and determination of ways to solve the problem</i> |
| 8 - 11 | <i>Execution of the programmatic part of the Coursework</i> |
| 12 - 13 | <i>Software Product Testing</i> |
| 14 - 15 | <i>Design of the explanatory note to the Coursework</i> |
| 16 | <i>Submission of Coursework for verification</i> |
| 17 | <i>Defense of Coursework</i> |

6. Self-study

| # in/o | Names of topics that are submitted for independent study | Number of hours of self-study |
|---------------|--|--------------------------------------|
| 1 | <i>Getting a topic and assignment for Coursework</i> | 1 |
| 2 | <i>Analysis of the task, selection and study of literature</i> | 5 |
| 3 | <i>Execution of the programmatic part of the Coursework and it testing</i> | 20 |
| 4 | <i>Design of the explanatory note to the Coursework</i> | 4 |
| | Total: | 30 |

Policy and Assessment

7. Course policy

Design of the work must meet the requirements for reports on research works (ДСТУ 3008-2015 «Державний стандарт України. Документація. Звіти в сфері науки і техніки. Структура і правила оформлення»).

All illustrative material in Coursework must be done using computer tools. The content of the illustrative material should fully reflect the main provisions that are submitted for defense.

Both the teacher and the student are required to abide by the «Igor Sikorsky Kyiv Polytechnic Institute» University Codex of Honor.

Course Policy Highlights:

- the topic of the Coursework can be coordinated with the topic of the future qualifying work of the bachelor;
- stages of the Coursework must be performed in accordance with the calendar schedule of work;
- the developed software product should be tested on a variety of control examples, the results of which are given in the text of the Coursework;
- in case of revealing the fact of academic dishonesty and plagiarism, the Coursework is returned for a radical alteration with a possible replacement of its topic;
- untimely completion of the work stage entails a decrease in the points received for it by 10%, if the delay is less than two weeks, by 20% if the delay is more than two weeks.

When evaluating Coursework, next consideration is taken into account:

- completeness of the individual task for Coursework;
- operability of the developed software product;
- timeliness of Coursework according to the schedule;
- independence of Coursework and no indications of plagiarism;
- answers to questions on the content of the work during its defense.

8. Monitoring and grading policy

The system of assessing the grading policy in the discipline "Coursework on System Programming" is based on the "Regulations on the system of assessment of learning outcomes in the «Igor Sikorsky Kyiv Polytechnic Institute» (https://document.kpi.ua/files/2020_1-273.pdf), namely the Rating System of Assessment (RSA) of the second type (RSA-2).

RSA-2 of Coursework R_C consists of two components:

- starting (R_S);
- component of defense (R_D).

$$R_C = R_S + R_D$$

The first (starting) component characterizes the student's work on the Coursework and its result - the quality of the explanatory note and developed by Software. The second component characterizes the quality of defense by the student of the Coursework.

The size of the scale of the first component is **80 points**, and the second component - **20 points**.

Quality of the explanatory note and degree of compliance with the calendar work schedule

Weight score – **80 (R_S)**. Criteria for evaluating the individual components of the explanatory note are given in the Table 8.1.

Table 8.1 – Criteria for evaluating the implementation of individual components of the explanatory note

| No of stage | Components of Coursework | Maximum number of points with timely execution | Taking into account the timeliness of execution |
|--------------------|---|---|--|
| 1 | Cover page design | 2 | 100% 100% of the assessment while adhering to the work |
| 2 | Availability of Coursework job sheet | 2 | |
| 3 | Availability and content of the abstract | 2 | |
| 4 | Availability of Index | 2 | |
| 5 | Availability and content of introduction | 2 | |
| 6 | Availability and content of problem statement | 5 | |

| | | | |
|----|---|-----------|--|
| 7 | Availability and content of the description of methods of solving the problem | 15 | schedule 90% in case of delay is less than 2 weeks 80% in case of delay is more than 2 weeks |
| 8 | Availability and content of the description of the software product | 15 | |
| 9 | Availability and content of the software product test results | 15 | |
| 10 | Availability and content conclusions | 10 | |
| 11 | Availability and design of references | 10 | |
| | Total R_S | 80 | |

The student is allowed to defend the coursework, provided that he has a starting component R_c of not less than 60% of the maximum value, which is

$$80 \times 0.6 = 48 \text{ points.}$$

Quality of defense

Weight score – **20 (R_D)**.

Criteria for evaluating the presentation on the materials of the CW and the answers to questions:

- degree of proficiency in theoretical material: up to 10 points;
- degree of proficiency in the program code: up to 10 points.

The defense of the coursework is considered successful if the R_D is not less than 60% of the maximum value, which is

$$20 \times 0.6 = 12 \text{ points.}$$

after the work protection is completed, R_K is determined, with further conversion of grades into a university scale in accordance with the table:

| Score R_K | Grade |
|---------------------------------|--------------|
| 100-95 | Excellent |
| 94-85 | Very good |
| 84-75 | Good |
| 74-65 | Satisfactory |
| 64-60 | Sufficient |
| Below 60 | Fail |
| Course requirements are not met | Not Graded |

Syllabus of the course

Is designed by teacher PhD, Associate Professor, Valerii Pavlov

Adopted by Department of Computer Engineering (protocol #10 , May 25, 2022)

Approved by the Faculty Board of Methodology (protocol #10 , June 09, 2022)