UDC 004.021+378.147

N.V. Karpenko, L.Y. Martynovych

THE FEATURES OF PROGRAMMING TEACHING AT THE UNIVERSITY

Abstract. The assimilation of programming by the students of Dnipropetrovsk National University (majoring in Computer Engineering) was analyzed. The basic problems that the students face in programming were defined. The correlation coefficient of success in students training in various disciplines was compared.

Keywords: programming teaching, solving tasks, successful training and the coefficient of correlation.

Introduction. The national market of information technologies is under active development in the XXI century. In particular, the level of computerization, Internet expansion rates, and the export of Ukrainian software products were increasing [1].

According to research of Ukraine Digital News and Fund of AVentures Capital the total number of programmers in Ukrainian IT market is 90 thousand people. Analysts are forecasting the growth of the number of programmers and software professionals up to 200 thousand persons by 2020. Today the demand for software developers in Ukraine is more than supply [2]. However, Ukrainian specialists must be competitive in the labor market in developed countries. Thus, the important role of the university is to improve the quality of training of the IT specialists.

The aim of this work is to analyze mastering the basics of programming and determine the main problems the students face during their training.

Literature review. IT-specialist often solves tasks based on analysis and mathematical concepts. Therefore, this specialist must have a basic level of mathematical training. The programmer also should have algorithmic thinking. However, the most of publications dedicated to the discussion on choices of a programming language, avoiding the aspects of teaching [3]. In this regard, the "Cone of Experience" proposed

 ISSN 1562-9945

by Edgar Dale [4-5] seems very interesting and useful (Fig. 1). During the 1960s, Edgar Dale assumed that learners retain more information by what they "do" as opposed to what they "heard", "read" or "observed". Today this "learning by doing" has become known as "experiential learning" or "action learning".

In other words, the lower the level of abstraction of material to be learned, the more effective results will be achieved.

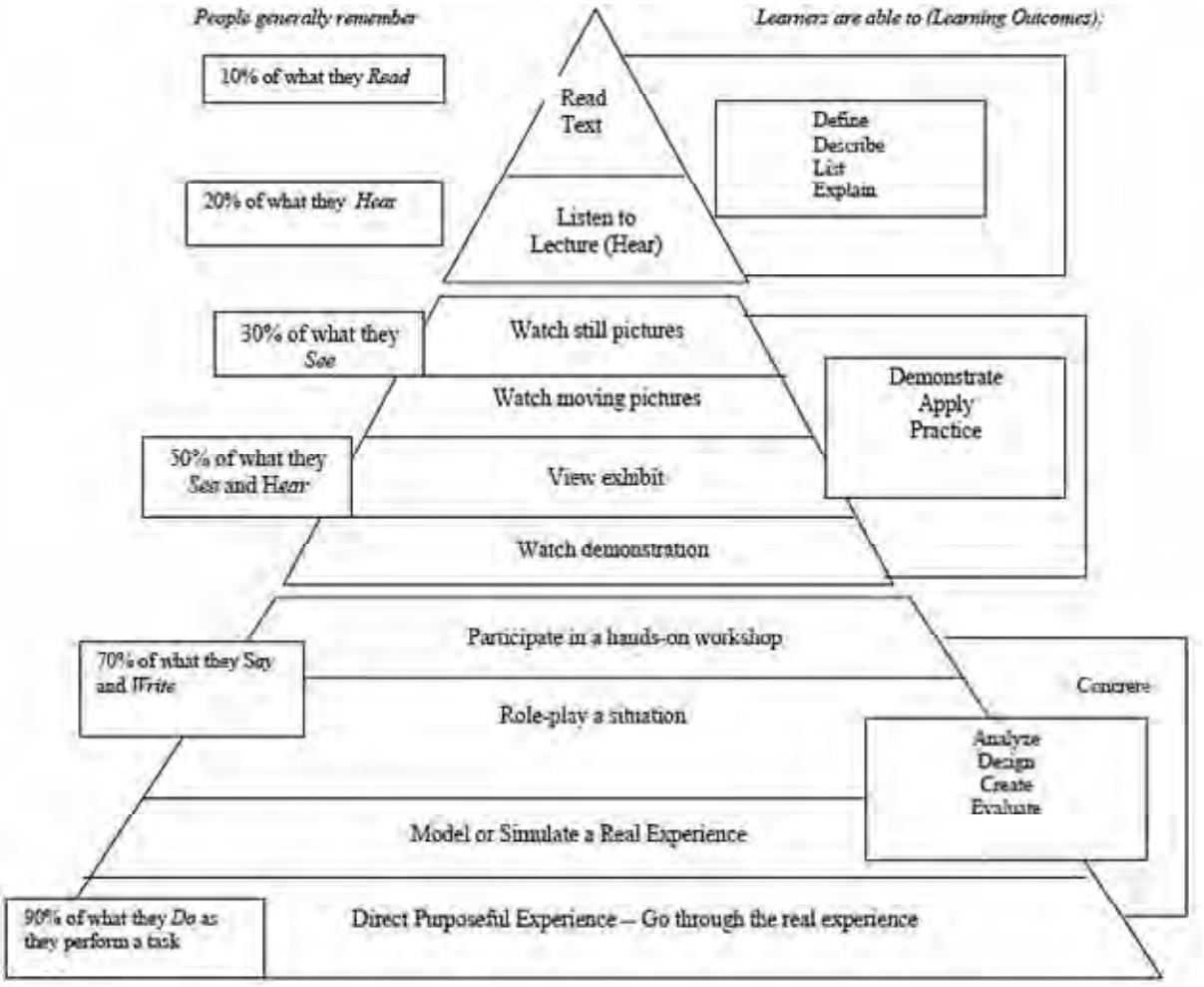

Figure 1 – Cone of experience offered by Edgar Dale [5]

Main part. After analyzing the data presented at http://www.vstup.info, we can generally understand the abilities of future students. For example, we can see the score of UPE and the school certificate score etc. However, it does not give us information on their abilities of logical thinking, i.e. abilities to recognize analogies and generalize the data, abilities of reasoning and proof?

Each year during the practice in programming we face the fact that students don't see and cannot keep track of regularity in solving

tasks. So, the teacher must find tasks for the development of these skills. For example, the students must propose the solution for finding the proper ending of word (Tab. 1). Consequently, they should track natural numbers that give different endings of the word (машин_, машин**а**, машин**и**).

Table 1

Determining the regularities in occurrence of certain endings of words

The word / The regularity	«машин_»	«машин**а**»	«машин**и**»
The natural numbers leading to appropriate ending of word	0, 5,6,7,8,9	1	2, 3, 4
An exception	11, 12,13, 14		

Thus, the condition for solving the problem about the number of cars will take the following form:

```
remain=car%10;
if(remain==1 && car!=11)
        printf("%d машина\n",car);
else if (remain ==2 || remain==3 || remain==4 && car!=12 &&
        car!=13 && car!=14)
        printf("%d машини\n",car);
else printf("%d машин\n",car);
```

Another problem that is difficult for students is to *determine the minimum time that passes before the hour and minute hands become perpendicular to each other*. The students must remember that full circle covers 360°. Then the phrase "the hour and minute hands are perpendicular to each other" should be written as the difference in absolute values (modulus): $\left|\phi_{\text{hour}} - \phi_{\text{minute}}\right| = 90^\circ$, where $\phi_{\text{minute}} = \frac{360^\circ}{60} \cdot m$ and $\phi_{\text{hour}} = \frac{360^\circ}{12} \cdot h + \frac{360^\circ}{12 \cdot 60} \cdot m$; h – number of hours, m – number of minutes ($0 \le \boldsymbol{h} \le 11$; $0 \le \boldsymbol{m} \le 59$).

The major problem of first-year students is inability to work independently when they learn subjects beyond the tutorial, to acquire optional theoretical knowledge and practical skills systematically. They try to study the subject just before the exam. But it does not work for the subjects that require practical experience.

Analyzing GPA school certificate, we have an idea to compare the overall level of students training depending on entering year (Fig. 2, a).

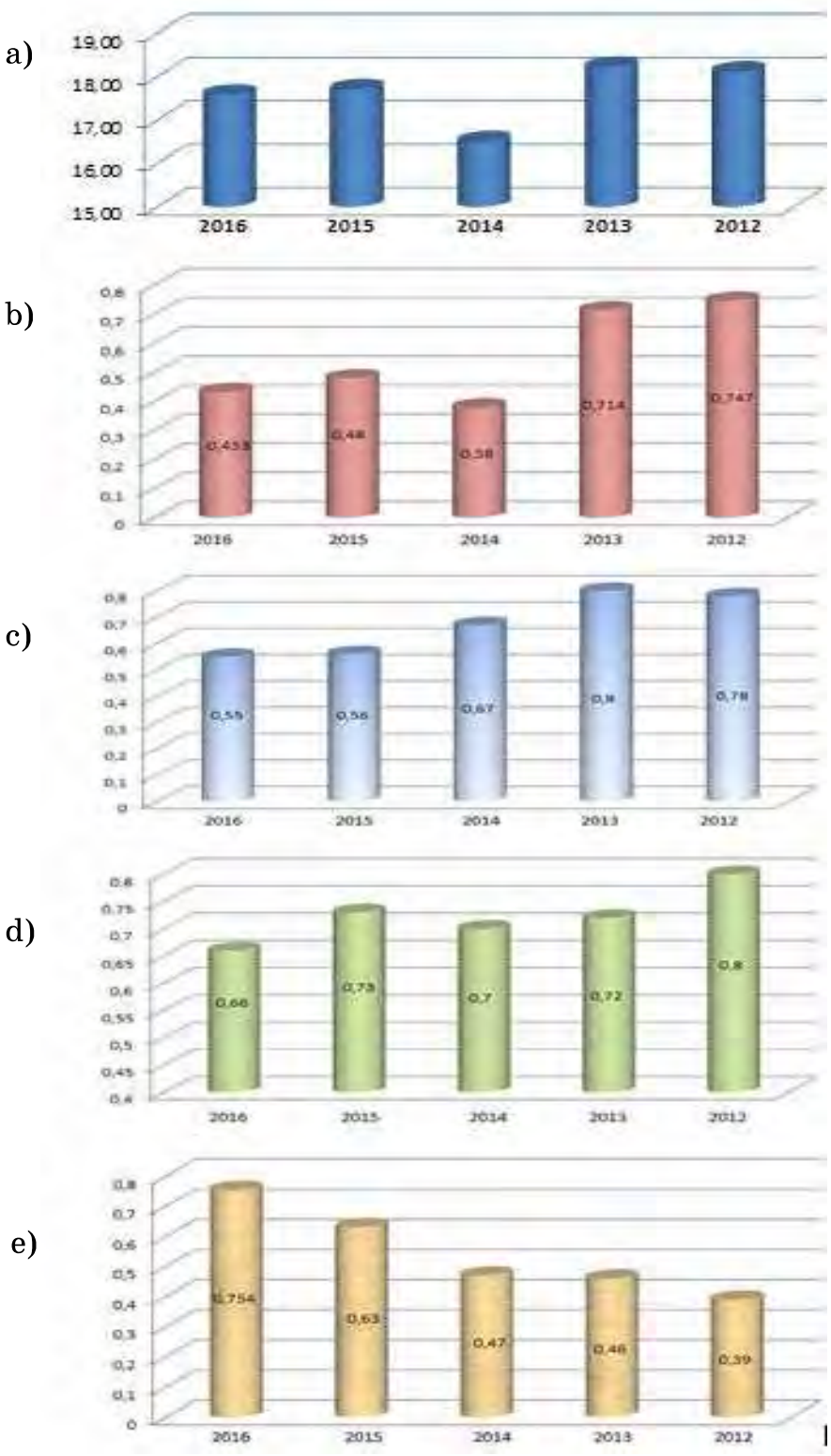

Figure 2 – Comparison of the successfulness in training students of different grade levels, the average mark of school certificate (a) as well as the correlation between the average school mark and average midyear examinations mark (b) for the following disciplines: "Mathematics - Programming" (c); "Physics-Programming" (d); «Web-design Programming" (e)

We can track the correlation between the average mark of the school certificate and the average mark that student receives during the first midyear examinations (Fig. 2, b).

The correlation between these indices could be determined for 2012-2013, but since 2013 we haven't determined a sustainable relation. A comparison shows the coefficients of correlation between the different disciplines delivered to the students during the first semester (Fig. 2, c-e). The relationship between the *Higher Mathematics* and *Programming* subjects is shown in Fig. 2, c. This coefficient of correlation has been decreasing during last three years. It happened because the students think of *Higher Mathematics* as of an abstraction far from satisfying the real-life needs, and therefore they are incapable to apply the knowledge and skills acquired to solve the programming problems.

The histogram (Fig. 2,d) shows a good correlation between the *Physics* and *Programming* subjects. The close relationship between these disciplines can be explained by the fact that students study the various natural phenomena experimentally. It develops the skills in analysis and synthesis. The students get a useful experience. It values highly for learning programming.

The histogram (Fig. 2,e) shows the distribution of correlation coefficients between the *Web-design* and *Programming* disciplines. In the last two years, the similarity of the students' marks is observed. This result may be explained by the fact that the first-year students (2015-2016 years) are more creative than the senior students. Therefore, it would be a good practice to consider students' abilities in planning the annual workload. The planning should be flexible. For example, the number of disciplines associated with individual workload and creativity (visual programming, web-design etc.) should be increased for practical-result oriented creative students. But the curriculum should not be the same for analytically thinking person able to achieve the best results in system modeling and engineering, quality assurance etc.

Conclusions. The starting point for the software specialist's education is testing and, depending on the testing results, development of their analytical capabilities. It can be achieved by reducing the high-level abstraction material quota and increasing the number of practical lessons in the discipline.

REFERENCES

1. Exploring Ukraine. IT Outsourcing Industry 2012 [Electronic resource]. – Access mode : http://hi-tech.org.ua/wp-content/uploads/2012/08/
2. Exploring-Ukraine-IT-Outsourcing-Industry-20121.pdf.
3. Шадий О. Рынок информационных технологий в 2016 году: путь через тернии [Electronic resource]. – Access mode : http://racurs.ua/1070-rynok-informacionnyh-tehnologiy-v-2016-godu-put-cherez-ternii.
4. Овчаров С. А. Особенности преподавания программирования бакалаврам бизнес-информатики. /Известия Волгоградского государственного технического университета. – 2013. – №16 (119). – Т. 14. – с. 94-95.
5. Как эффективно учиться? [Electronic resource]. – Access mode : http://www.insai.ru/statya/kak-effektivno-uchitsya.
6. Dale, Edgar. Audio-Visual Methods in Teaching, 3rd ed., Holt, Rinehart & Winston, New York, 1969, p. 108.