Atividade - Linking Words

Discentes

Igor Lima Rocha

Q1 - Analise o texto e responda

a) **F**
b) **F**
c) **F**
d) **F**
e) **T**

Q2 - Linking Words

In general	-	**No geral**
Otherwise	-	**De outra forma**
Finally	-	**Finalmente**
As a result	-	**Como resultado**
However	-	**No entanto**
Moreover	-	**Além disso**
Though	-	**No entanto**
Such as	-	**Assim como**
In addition	-	**Além disso**
Such	-	**Como**

Q3 - Traduzir palavras

a) operations - **Operações**
b) importantly - **De forma importante**
c) conditional - **Condicionalmente**
d) essentially - **Essencialmente**
e) instruction - **Instrução**
f) predetermined - **Predeterminado**
g) completeness - **Completude**
h) simultaneously - **Simultaneamente**
i) programming - **Programação**
j) subfields - **Subcampos**
k) overlaps - **Sobreposições**
l) bioinformatics - **Bioinformática**
m) interdisciplinary - **Interdisciplinaridade**

Q4 - Incluído na linguagem do computador

Computer languages include conditional instructions - essentially, rules that say, “if memory location n satisfies condition a, do instruction number x next, otherwise do instruction y.”

Q5 - Loop

Sequences of instructions that are repeated a number of times until a predetermined condition becomes true.

Q6 - Subáreas da computação

The major subfields of computer science include the traditional study of **computer architecture**, **programming languages**, and **software development**. However, they also include **computational science** (the use of algorithmic techniques for modeling scientific data), **graphics and visualization**, **human-computer interaction**, **databases** and

information systems, **networks**, and the social and professional issues that are unique to the practice of computer science.

Q7 - Scanning

a) bloqueio e desbloqueio de registros: **locking and unlocking of records**
b) ambiente multiprogramação: **multiprogramming environments**
c) ambiente multiusuário: **multi-user environments**
d) ponto de recomeço: **restart point**
e) ambiente multitarefa: **multi-tasking environments**

Q8 - Associar expressões

work scheduling - **escalonamento de tarefas**

shared access - **acesso compartilhado**

background job - **tarefa secundária**

foreground job - **tarefa principal**

backing storage - **memória secundária**

Q9 - Significado

Multi-tasking environment

The running of multiple programs (sets of instructions) in one computer at the same time.

Multi-user environment

An environment where other users can connect and make changes to the same database that you are working with.

Batch x Real-Time environment

Batch mode means that a ‘background’ job deals with programs that run without any terminal or user interaction. Real-time jobs, on the other hand, are ‘foreground’ jobs that

deal with situations in which immediate results are necessary – such as airline seat reservations, on-line booking of hotel accommodations, automatic cash dispenser, etc.

Q10 - Sistema operacional

Multi-tasking environment

Requires a powerful operating system incorporating work scheduling facilities to control the switching between programs.

Multi-user environment

An operating system is required to control terminal operations on a shared access basis and also the locking and unlocking of records to prevent one user attempting to read a record while another user is updating it, for instance.

Batch x Real-Time environment

The operating system:

a) interrupts batch processing to deal with real-time inquiries, and
b) transfers the interrupted stage of batch processing to backing storage.

Q11 - Completar a tabela

NOUN	VERB	ADJECTIVE	ADVERB
product	produce	productive	productively
automation	automate	automated	automatically
computer	compute	computed	—
interaction	interact	interactive	interactively
operation	operate	operational	operationally
interruption	interrupt	interrupted	—

Q12 - Cognatos

The Universal Turing Machine

In 1936, at Cambridge University, Turing invented the principle of the modern computer. He described an abstract digital computing machine consisting of a limitless memory and a scanner that moves back and forth through the memory, symbol by symbol, reading what it finds and writing further symbols (Turing [1936]). The actions of the scanner are dictated by a program of instructions that is stored in the memory in the form of symbols. This is Turing's stored-program concept, and implicit in it is the possibility of the machine operating on and modifying its own program. (In London in 1947, in the course of what was, so far as is known, the earliest public lecture to mention computer intelligence, Turing said, 'What we want is a machine that can learn from experience', adding that the 'possibility of letting the machine alter its own instructions provides the mechanism for this' (Turing [1947] p. 393). Turing's computing machine of 1936 is now known simply as the universal Turing machine. Cambridge mathematician Max Newman remarked that right from the start Turing was interested in the possibility of actually building a computing machine of the sort that he had described (Newman in an interview with Christopher Evans in Evans [197?].

From the start of the Second World War Turing was a leading cryptanalyst at the Government Code and Cypher School, Bletchley Park. Here he became familiar with Thomas Flowers' work involving large-scale high-speed electronic switching. However, Turing could not turn to the project of building an electronic stored-program computing machine until the cessation of hostilities in Europe in 1945.

During the wartime years Turing did give considerable thought to the question of machine intelligence. Colleagues at Bletchley Park recall numerous off-duty discussions with him on the topic, and at one point Turing circulated a typewritten report (now lost) setting out some of his ideas. One of these colleagues, Donald Michie (who later founded the Department of Machine Intelligence and Perception at the University of Edinburgh), remembers Turing talking often about the possibility of computing machines (1) learning from experience and (2) solving problems by means of searching through the space of possible solutions, guided by rule-of-thumb principles (Michie in interview with Copeland, 1995). The modern term for the latter idea is 'heuristic search', a heuristic being any

rule-of-thumb principle that cuts down the amount of searching required in order to find a solution to a problem. At Bletchley Park Turing illustrated his ideas on machine intelligence by reference to chess. Michie recalls Turing experimenting with heuristics that later became common in chess programming (in particular minimax and best-first).

Further information about Turing and the computer, including his wartime work on code breaking and his thinking about artificial intelligence and artificial life, can be found in Copeland 2004.