

NHẬP DỮ LIỆU TỪ BÀN PHÍM

Nhập tử tế, đúng yêu cầu	Nhập ngoài biên (out of range), ngoài miền giá trị cần thiết Dùng if	Lệnh nhập bị trôi, lệnh lạc trôi (Skip getting a char/string)	Nhập cà chớn, không biết xử lí ra sao Không xài if được (Exception)
<p>App yêu cầu nhập chuỗi</p> <ul style="list-style-type: none"> Ta gõ "Ahihi" rồi Enter Ồn, xử lí tiếp không bình luận <p>App yêu cầu nhập số nguyên</p> <ul style="list-style-type: none"> Ta gõ 2000 rồi Enter 	<p>App yêu cầu nhập năm sinh từ [1990...2001]</p> <ul style="list-style-type: none"> Ta gõ 2000 Enter > Ổn Ta gõ 2005 Enter > giá trị không hợp lệ, nằm ngoài biên, ta dùng lệnh <code>if(yob < 1990 yob > 2001)</code> <code>sout(chữ) ;</code> Và có thể kết hợp do-while() bắt nhập lại 	<p>Xảy ra khi nhập kí tự/chuỗi sau khi nhập số (lệnh nhập số trước lệnh nhập chữ/chuỗi)</p> <p>Nguyên nhân:</p> <ul style="list-style-type: none"> Lệnh nhập số chỉ dùng đúng phần số trong buffer (bộ nhớ đệm nhập xuất), và để lại dấu cách, kí tự khác, và enter trong buffer (gọi là rác buffer). Lệnh liền kề nhập kí tự/chuỗi thấy có "char" trong buffer, hốt luôn "rác" này gán vào biến kí tự/chuỗi 	<p>Xảy ra khi yêu cầu nhập số nhưng ta gõ chuỗi Ahihi</p> <ul style="list-style-type: none"> Mặc định, JVM phát hiện dữ liệu sai lệch, không thể convert "Ahihi" về số nào phù hợp, để đảm bảo App xử lí data an toàn, JVM gửi một "tin nhắn"/an Exception object cho App, báo rằng có tình huống "bất thường, cà chớn không như dự kiến – gọi chung là Exception" đã xảy ra, và giết luôn App, ngay và luôn (End Process)

<ul style="list-style-type: none"> ○ Ổn, chơi tiếp 	<p>Hoặc App yêu cầu nhập chuỗi khác rỗng, theo định dạng nào đó</p> <ul style="list-style-type: none"> ○ Ta dùng <code>if()</code> và các hàm liên quan để kiểm tra dữ liệu nhập vào và thông báo lỗi (và bắt nhập lại) là ổn ○ Nhập cùng data type khác kì vọng, xài <code>if</code> ○ Đúng định dạng, sai value cần -> ngoài biên -> if mà check mà xử 	<ul style="list-style-type: none"> ○ Và hiện tượng trôi lệnh xảy ra, cảm giác lệnh nhập kí tự/chuỗi không chạy, không cho mình gõ nhưng thực ra là “tài lạnh” hốt rác buffer gán vào <p>Lập trình C: xóa bộ nhớ đệm nhập xuất trước lệnh nhập kí tự/chuỗi %c, %s</p> <ul style="list-style-type: none"> ○ <code>fflush(stdin)</code> ; nếu xài compiler MinGW, Microsoft C Compiler ○ <code>fpurge(stdin)</code> ; nếu xài GCC, Cygwin Compiler ○ <code>__fpurge(stdin)</code> ; nếu xài Linux, MacOS <p>Lập trình Java: xóa bộ nhớ đệm nhập xuất trước lệnh nhập kí tự/chuỗi dùng Scanner</p> <ul style="list-style-type: none"> ○ Tà chiêu Âu Dương Phong <code>sc.nextLine()</code> ; chạy chơi ○ Chánh đạo Quách Tĩnh <code>Integer.parseInt(sc.nextLine());</code> <code>class TraiBao, Wrapper</code> <code>class (boxing/unboxing)</code> 	<ul style="list-style-type: none"> ○ Lí do: đưa chuỗi vào số, không thể dùng lệnh <code>if</code> để xử lí được, mà tiếp tục dùng cái sai để làm tiếp thì là sai luôn, giết cho lành ○ Nếu không muốn bị JVM giết khi có “cà chớn”, hãy ra hiệu với JVM đừng giết App, cứ nhắc App có cà chớn là được, gửi cho App cái tình hình cà chớn – an Exception object, và để App tự tính toán, tự xử. JVM sẽ không giết App nữa, nhưng JVM muốn rằng App cần có cơ chế cam kết đảm bảo xử lí cà chớn. JVM muốn thấy App phải nhốt các lệnh có nguy cơ gây cà chớn vào tầm ngắm, vào tầm kiểm soát, giám sát, ĐÓ LÀ CƠ CHẾ TRY-CATCH <pre>try { nhốt lệnh cần giám sát .nextInt(); .parseInt(?); } catch(Exception e) { e chính là đối tượng, tình hình cà chớn mà JVM nắm bắt được gồm chi tiết nguyên nhân gây ra cà chớn. Hãy ex. chăm đề xem sự vụ diễn ra</pre>
---	---	--	---

			thêm các lệnh xử lí hậu quả ở đây }
--	--	--	---

KHI NÀO XÀI TRY-CATCH

<ul style="list-style-type: none"> Convert data không thành công (chuỗi - > số) Chia cho 0 <p>Mà ko chặn số chia qua if</p>	<ul style="list-style-type: none"> Đi vượt quá số phần tử mảng (out of bound) Null Pointer – Chưa gán giá trị cho biến con trỏ/biến object mà bày đặt đòi chấm 	<ul style="list-style-type: none"> Kết nối mạng, truyền nhận dữ liệu qua mạng, database ở xa. Server chết, mạng rớt <p>Sql injection</p>	<ul style="list-style-type: none"> I/O nhập xuất với tập tin. Bắt buộc phải nhốt lệnh vào trong try-catch Còn nhiều tình huống khác nữa...
--	--	---	--