1) What is BPR really? (Not “improve tasks” — redesign reality)

Your teacher’s opening idea is gold:

Idea → Reality → Application

An **idea** is defined by the **attributes/elements of reality** that make it true. “Understanding an idea” means you can **apply it properly in new situations** (not just repeat definitions). That’s why he keeps pushing you to take *any* process (enrollment, Starbucks, NADRA, tea, airport boarding) and reengineer it.

BPR definition (your wording + clean exam wording)

Business Process Reengineering = taking a core process, identifying what must *essentially* be achieved (the value/constant), and then radically redesigning the steps + rules + data + responsibilities to achieve dramatic improvement in performance (cost/time/quality/productivity/customer value).

This connects directly to the course outline’s “as-is → diagnose → to-be redesign, with tech enablers + change management”.

---

2) Process = Rules applied on Data (the “data lens” that your sir loves)

This is one of his signature lines:

“Every process is no more than rules applied on data.”

A process step can only happen if the **required data exists** (or is verified). So when you “walk through a process,” you are basically **traveling through data states**.

Example: UMS enrollment (your note)

- If system already has: *core requirements, courses taken, prerequisites, fee status, ERP/student profile*
  Then why show a student courses they **cannot** take?
  A data-driven redesign would show only eligible options (rule: `eligible(course, student_data)`).

That's exactly what "data-driven process engineering" means in your notes: **engineer the data and the rules**, not just the UI. (And this is basically the "don't automate, obliterate" logic: don't digitize a stupid flow; delete the stupidity.)

---

3) Why "Don't automate, obliterate" is the whole soul of BPR

Your teacher's argument:

- A process might have been "perfect" for its era.
- Society/infra/tech changes → the process becomes **redundant**.
- So incremental automation just makes a redundant process faster (still redundant).

Exam phrasing:
"Automating a broken or outdated workflow increases speed but preserves waste. BPR asks whether the step is still necessary under today's infrastructure and data availability."

This idea appears again later as **unlearning** (Section 8).

---

4) Why processes exist: Value, and why value needs consistency

Your "economics detour" is not random — it's the philosophical base:

Macro vs Micro (why firms matter)

- **Micro (firms)** generate wealth (real value creation).
- **Macro (state)** distributes wealth (taxes, policy).
  If you have no micro/production base, the economy is "dead," because financial instruments don't feed people.

What is value?

Value = delivering a product/service society needs/wants/benefits from.

Why process matters

Your teacher's one-word answer:
Companies consistently generate value through PROCESS.
Because process is how you deliver the *same output* with predictable quality.

Milk example: we pay not for "milk" but for **consistent value** (same taste every time). That consistency is process control.

---

5) Efficiency vs Productivity (and why "efficient" can still be bad)

You wrote: "a process designed for efficiency may not yield productivity."

Clean distinction

- **Efficiency**: using fewer resources per unit (doing it with less waste).
- **Productivity**: output per time/resource (doing more per same input).

You can be efficient but produce little (tight controls, slow throughput), or productive but inefficient (high output but lots of waste/cost). BPR decides what you're optimizing using a **merit function**.

---

6) Merit function (your sir's favorite mathematical "thinking tool")

He's basically teaching you: "Stop arguing emotionally. Make the process measurable."

Core structure

You define steps: (s_1, s_2, …, s_n).
Each step has a measure (m_i) (time, error rate, queue length, cost, etc.).
Then you define an objective:

Merit Function:
[
F = \sum_{i=1}^{n} w_i \cdot m_i
]
where weights (w_i) represent **importance** of that step/metric.

This exact “weighted sum of measures” idea shows up in your 29 Nov notes about heterogenous measures and weights.

The “exam trap”: heterogeneous measures (queue time vs CSAT)

Your teacher calls this out as a trap:

- **Queue time**: objective minutes
- **CSAT**: numeric but based on subjective feelings

You cannot just add “12 minutes + 4.2 stars” raw. You must:

1. **Normalize / scale** (convert to comparable scale like 0–1)
2. Use **weights carefully**
3. Explain tradeoff: “fast but rude” can reduce CSAT even if time improves.

That’s exactly your written point and it’s extremely testable.

---

7) Process mapping + timestamps (why “flowcharts” aren’t art, they are data)

Your sir keeps forcing you to go **micro** (“smallest steps possible”) because if steps are fuzzy, you can’t measure them.

A good process map has:

- clear steps
- clear owner (who does it)
- clear data inputs/outputs
- timestamps / durations

He even says: “A process has time stamps.”
And he pushes you to reflect the flowchart into an **Excel simulation** (cells = time/events), run it, and extract insight.

That’s also the bridge to **process mining** later.

---

8) Tools to diagnose broken processes (before redesign)

Fishbone (Ishikawa)

It forces you to categorize causes so you don't miss hidden drivers. Your tea example breaks causes into buckets like:

- Measurements
- Materials
- People
- Environment
- Machines
- Process
- Problems

Your Six Sigma tea notes explicitly do this and explain how tiny variations add up to failures.

5 Whys

Use it to reach root cause, not symptoms:

- "Why is tea late?" → "heater slow"
- "Why heater slow?" → "maintenance/no preheating"
- "Why no preheating?" → "no role/standard work"
  …and so on.

Pareto

80/20: identify the few causes creating most delay/defects.

BPR logic: Diagnose first → only then redesign the "to-be".

---

9) Six Sigma in this course (not stats for fun — it's process consistency)

Your teacher's framing:

- A process is meaningful only if it's **measurable**.
- Variation = enemy.
- Sigma = standard deviation → shows spread.

What you did in tea example

- requirement: tea within 5 minutes
- you computed mean, SD, sigma level, defect rates etc.
- conclusion: low sigma (bad capability), too many late teas.

The writeup explains: many small delays (missing spoon, heater, missing cups) accumulate into late service — exactly what Six Sigma tries to squeeze out.

“Every failure is rework”

Your sir’s point: defects force the process to rerun or compensate → **hidden cost**.

Defects per million (DPMO logic)

He’s testing whether you can translate yield ↔ defects ↔ sigma thinking. Your notes show that for a high-quality process, defects must be extremely low; for others (chairs) tolerance is higher.

Key exam phrasing

“Six Sigma is a measurement and control philosophy: define defects, define opportunities, collect time/quality data at each step, analyze variation, and redesign/control the process so defect probability becomes extremely small.”

Bonus: “Six Sigma works when data capture is automatic”

Your teacher keeps repeating this — because manual measurement is expensive and inconsistent. That’s why he keeps saying tech enables measurement/control.

---

10) Process Mining (DISCO) — the “event log lens” vs data mining

You wrote the clean distinction:

Process mining vs data mining

- **Process mining has timestamps + event sequences** (case flows).
- **Data mining** can analyze data but doesn’t inherently reconstruct “the process story.”

From your notes:

- A **case ID** = one full journey from start to end.
- **Events/activities** = steps.
- A **variant** = a unique path pattern (deviation).
- The most common variant is the “happy path.”
- Too many variants (e.g., 608 cases and 608 variants) means poor standardization.

You also have the concrete Disco task:

- **6 events**: case **157**
- **18 events**: case **335**

What the 6 vs 18 events teaches (the BPR lesson)

- **18-event case (335)** = full procurement cycle, long duration, many handoffs/approvals, visible bottlenecks.
- **6-event case (157)** = short, ends early (stuck or cancelled), likely incomplete.

Your provided analysis explicitly flags: reduce waiting times, simplify supplier communication, automate reminders/approvals.

Exam-ready interpretation:

- BPR goal: reduce variants (standardize), reduce non-value steps (rework, repeated analysis), reduce bottleneck waiting times.
- Tech lever: automated escalations, approval rules, dashboards.

---

11) Digital tools in BPR: when tech is a lever (and when it's a trap)

Your course explicitly includes "digital tools and BPR" and "process mining," and your teacher adds economic + governance angles.

Starbucks (perfect BPR + digital flywheel example)

Your uploaded Starbucks writeup is basically a model answer:

Before: in-person ordering + payment at counter → one physical queue bottleneck
After: Mobile Order & Pay:

- ordering outsourced to customer (off-site)
- pre-payment eliminates payment transaction time
- queue split into two streams (parallelization)
- loyalty + personalization creates a feedback loop (data → offers → more visits)

Objective functions achieved (exam language):

- reduce wait time (convenience)
- increase throughput (peak hour capacity)
- increase revenue/volume
- improve working capital (stored funds in app)

- improve order accuracy (digital customization)

Tesla Giga Press (from your outline)

This is BPR in manufacturing:

- replaces dozens of parts/welds with single casting
- reduces part count, tooling, time, footprint
  That's **radical redesign**, not incremental improvement.

---

12) Economics of BPR: ROI is not "only money," but money must be proven

Your teacher's ROI framing:

Step 1: Start with the merit function

You don't start with "profit." You start with: "what are we optimizing?"
Then you show how that relates to business outcomes (e.g., customer satisfaction → lower churn, higher repeat purchases).

CAPEX vs OPEX (very testable)

- **CAPEX**: upfront investment (kiosk setup, machines)
- **OPEX**: monthly running cost (maintenance, support, printing, staff)

He warns: companies tolerate CAPEX mistakes more than OPEX mistakes because OPEX continuously eats profit.

Kiosk / agent cost model (your uploaded sheet)

You have this logic in the file: before vs after costs, using:

- NA = number of agents
- CTC (cost-to-company) per agent
- NF flights/day, CF customers/flight, scaled by 30 for monthly
- After redesign: agent time halves, but OPEX added, CAPEX exists, risk factor included

Exam-ready sentence:
"BPR ROI must compare unit cost before and after, spreading labor cost across throughput; redesign often reduces labor time per customer but introduces CAPEX/OPEX and risk, so feasibility depends on volume and cost discipline."

---

13) Organization-wide BPR: process inventory + cost/benefit matrix + CDO logic

This is your 26 Nov lecture (very exam-focused):

The CEO sets a target (e.g., 5% cost reduction)

Then each department creates an **inventory of processes** and estimates improvement potential.

But: **just because cost reduction is possible doesn't mean you should do it** — because redesign requires training, change effort, complexity.

Cost vs Benefit matrix (how to prioritize)

- Low cost + high benefit → prioritize (best)
- High cost + high benefit → strategic big project
- Low cost + low benefit → okay if bandwidth
- High cost + low benefit → usually avoid

Digital Transformation governance (very testable)

Your later notes add:

- outsource vs in-house digital tech
- SLA challenges (service must not degrade)
- some companies become "IT company with a banking license"
- if you build DT capability, it must be multipurpose, not one-usecase
- propose a DT department + CDO responsible for efficiency/cost/productivity
- CEO is ultimately responsible for processes

That's basically "process ownership + operating model" in BPR, and it connects to change management.

---

14) Unlearning (obliterate) — the "human brain" barrier to BPR

This is not motivational كلام — it's a real BPR constraint.

Why unlearning matters

Humans cling to old forms:

- keyboard phones felt “necessary” until touch UI matured
- the “constant” is not “keyboard”; it’s “input method”
  BPR demands you identify the **true constant** (the essential value) and delete the rest.

Boarding pass example (your note)

Goal/constant: “verify passenger’s right to enter plane.”
Boarding pass is only one old mechanism. You can redesign verification entirely.

Baggage example (your note)

Constant: bag must travel + weight allowance must be enforced.
Redesign idea: shift responsibility to customer via self-service (kiosk/weigh station) — **free outsourcing** — agent time freed → productivity increases, but you pay CAPEX/OPEX.

Exam-ready phrasing:
“Unlearning is the ability to separate a process’s essential purpose from its historical form; BPR requires unlearning because radical redesign breaks habits, roles, and identity, triggering resistance.”

---

15) Case studies you MUST be able to narrate (as stories)

Even if your notes are messy, your exam will likely ask “Explain how X was reengineered and what tech enabled it.”

Here are the story templates you should memorize:

Ford Accounts Payable (classic)

Old: matching PO + receiving + invoice across departments → reconciliation hell.
New: shared database + eliminate reconciliation by ensuring “data match” earlier, so AP doesn’t do wasteful checking.
Tech role: shared data/integration.
BPR principle: remove non-value work, shift control upstream.

GM Order-to-Delivery

Problem: silo handoffs across departments cause delays, errors, and long lead times.
BPR: integrate order flow end-to-end, reduce handoffs, make data visible.

Siemens/Boeing / Product development at scale

Old: sequential stage gates → slow time-to-market.
BPR: parallel development, cross-functional collaboration, fewer late reworks.

Taco Bell

Redesign the business model: focus on core, outsource non-core, change operating model.

Branchless banking (M-Pesa / Easypaisa / JazzCash)

Old: branch visits, paperwork, limited access.
New: phone-based transactions + agent network.
BPR: value delivery shifts from “branch process” to “mobile + agents.”
Objective: financial inclusion + speed + reach.

Ultra-Compact “Exam Answer Frames” (copy these in your head)

A) Define BPR (3 lines)

BPR is radical redesign of end-to-end processes to achieve dramatic gains in cost, time, quality, productivity, or customer value. It starts from the process purpose (value/constant), not existing departments. It often uses digital tech to eliminate non-value work, reduce handoffs, and control variation.

B) Don’t automate, obliterate

Automation speeds up the current workflow but preserves waste if the workflow is outdated.
BPR first questions whether steps are still necessary under today’s data/tech/infrastructure, then deletes redundant checks and rebuilds the “to-be” flow around the real purpose.

C) Merit function + heterogeneous measures

We quantify a process by defining steps and attaching measurable metrics (m_i) (time/cost/errors/CSAT) with weights (w_i) to represent importance, giving (F=\sum w_i m_i). If

measures are heterogeneous (minutes vs CSAT), we must normalize/scale and justify weights; otherwise the aggregate is meaningless.

D) Six Sigma in one paragraph

Six Sigma reduces defects by reducing variation. We measure step times/outputs, compute mean and standard deviation (sigma), identify bottlenecks causing spread, and redesign/control the process so defects become extremely rare (low DPMO). It works best when data capture is automatic and failures (rework) are explicitly treated as cost.

E) Process mining keywords

Case ID = one complete journey; event log = activities + timestamps; variants = different paths. The most common variant is the happy path; many variants indicate poor standardization. We use Disco to find bottlenecks, rework loops, long waiting times, and then redesign to reduce variants and delays.

Negative effects of BPR

1. **Layoffs and job insecurity**
   BPR often removes steps/roles ("obliterate"), so people can lose jobs or fear losing them. That fear creates stress and resistance.
2. **Resistance and morale drop**
   Radical redesign disrupts routines and identity ("this is how we've always done it"). People feel threatened, undervalued, or confused → motivation and cooperation drop.
3. **Loss of tacit knowledge**
   When experienced staff leave (or processes change too fast), the organization loses "unwritten" know-how (shortcuts, exception handling, relationships), which can hurt performance.
4. **Short-term productivity dip**
   During transition: training, learning new tools, new responsibilities, and confusion can temporarily slow work and increase mistakes.
5. **Process disruption and service failures**
   If "to-be" is rolled out poorly, customers face delays, errors, and inconsistent service (especially when the old process is switched off too early).
6. **High implementation cost (CAPEX + OPEX)**
   BPR may require tech systems, consultants, integration, training, and maintenance. Even if long-term savings exist, the short-term investment can be heavy.
7. **Technology dependency and new risks**
   Digitizing/redesigning can introduce:

- system downtime risk
- cybersecurity/privacy issues
- vendor lock-in (if outsourced)
- SLA problems (service-level failures)

8. **Political conflict and power shifts**
   BPR changes who controls decisions and data. Departments can fight to protect budgets, authority, or headcount → internal politics can block success.
9. **Over-standardization can reduce flexibility**
   To reduce variance, BPR may create strict “happy paths.” That can hurt handling of exceptions, special customers, or complex cases.
10. **Wrong metrics = wrong redesign**
    If the merit function/weights are flawed (or you mix heterogeneous measures badly), you optimize the wrong thing: e.g., lower wait time but worse CSAT, quality, or compliance.
11. **Compliance/legal/ethical issues**
    Redesign that ignores regulations (finance/healthcare/education) can create audit failures, legal risk, or unfair outcomes.
12. **Failure risk is high**
    Because BPR is radical, if assumptions are wrong or change management is weak, the whole initiative can fail—wasting money and damaging trust.